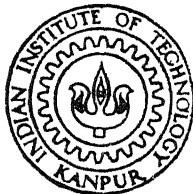


A QUEUING MODEL FOR SOME FLOW CONTROL SCHEMES IN COMPUTER NETWORKS

by
RAJA GHOSAL

TH
CSP/1982/14
G 346 9

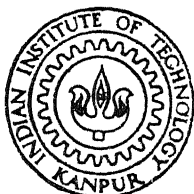


COMPUTER SCIENCE PROGRAMME
INDIAN INSTITUTE OF TECHNOLOGY, KANPUR
JULY, 1982

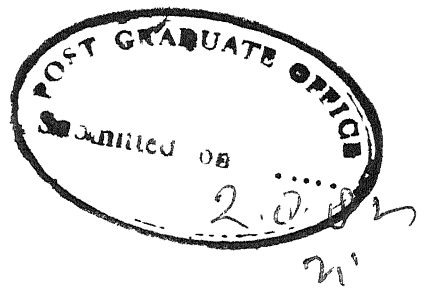
A QUEUING MODEL FOR SOME FLOW CONTROL SCHEMES IN COMPUTER NETWORKS

by
RAJA GHOSAL

TH
CSP/1982/14
G3469



COMPUTER SCIENCE PROGRAMME
INDIAN INSTITUTE OF TECHNOLOGY, KANPUR
JULY, 1982



CERTIFICATE

This is to certify that the thesis entitled
'A QUEUING MODEL OF SOME FLOW CONTROL SCHEMES IN COMPUTER
NETWORKS' has been carried out by Shri Raja Ghosal under
my supervision and has not been submitted elsewhere for the
award of a degree.

A handwritten signature in cursive script, which appears to read 'A.S. Sethi'.

Dr. A.S. Sethi
Assistant Professor
Computer Science Programme
Indian Institute of Technology, Kanpur

Kanpur
July, 1982

5 JUN 1984

CENTRAL LIBRARY

, K. n p j r

Acc. No A 83773

CSP-1902-M-GHO-SUE

ACKNOWLEDGEMENTS

I am deeply grateful to Dr. A.S. Sethi for his inspiring guidance, constant encouragement and perseverance throughout the course of this work. I am also grateful to him for readily being available for discussions and suggestions that were extremely useful.

I am thankful to the faculty members, Computer Science Programme, the Computer Center Staff and my fellow batchmates for making my stay at I.I.T. Kanpur most useful and pleasant.

I am thankful to K.S. Sateesh, P.K. Pandiya and G.P. Mattathil for their assistance in the preparation of this thesis.

I am thankful to Mr. C.M. Abraham for the efficient typing. He deserves special thanks for the enthusiasm and perseverance.

Raja Ghosal

ABSTRACT

Flow control in packet-switched computer networks tries to avoid undesirable effects like congestion, deadlocks and also aids in improving performance in the dynamic sharing environment. Flow control functions are defined at various levels of protocol hierarchy. The efficiency of the scheme at a given level is limited by the efficiency of the schemes below it. At the lowest level are link control schemes, which have to account for random factors like noise on the links. Existing open queuing models are unable to account for the window effect while closed queuing models are unable to account for retransmissions due to nodal buffer overflow. A 2-stage queuing model, suitable for analysis of most schemes is presented in this thesis. The parameters of interest are calculated numerically. A simple approximation, to account for the underestimation of delay by Markovian assumptions under moderate traffic is introduced. Simulation results are also presented to test the accuracy of the model.

TABLE OF CONTENTS

	Page
CHAPTER 1 FLOW CONTROL IN COMPUTER COMMUNICATION NETWORKS	
1.1 What is Flow Control	1
1.2 Why Packet Switching Allows Better Control	2
1.3 Objectives of a Flow Control Schema	5
1.4 Resources	5
1.5 Performance Measures	6
1.6 Levels of Flow Control in Data Networks	8
1.7 Flow Control Methods used at Various Levels of Protocols	11
1.8 Flow Control for Voice and Integrated Voice and Data Network	14
1.9 Flow Control in Interconnected Networks	16
- 1.10 Flow Control for Real-Time Communications	17
1.11 The Aim of This Study	17
CHAPTER 2 PERFORMANCE EVALUATION IN COMPUTER NETWORKS	
2.1 Introduction	20
2.2 Simple Queues	21
2.3 Queues with Intermediate Complexity	24
2.4 Complex Queuing Systems	31
2.5 Approximation Methods	33
2.6 Output Processes and Network of Queues	34
2.7 Applications of Network of Queues	39

	Page
2.8 Zero Insertion and its Influence on Packet Lengths for HDLC Protocols	41
2.9 Dynamic Flow Control	43
2.10 Flow Control in Integrated Voice and Data Networks	43
2.11 Proposed Queuing Model	44
CHAPTER 3 A QUEUING MODEL FOR SOME FLOW CONTROL SCHEMES	46
3.1 Introduction	46
3.2 Continuous Transmission with Infinite Window	53
3.3 Send-and-Wait Schemes	55
3.4 Continuous Transmission with Finite Window and No Error Recovery	57
3.5 Continuous Transmission with Finite Window and Retransmission with Finite Time-out	59
CHAPTER 4 DESIGN OF SIMULATION EXPERIMENT	
4.1 Runlength Calculations	63
4.2 Implementation of Simulation Model in PASCAL	63
4.3 Random Number Sequences used	70
4.4 Differences between Simulation and Analytical Model	71
CHAPTER 5 DISCUSSION OF RESULTS	
5.1 Accuracy of the Model	73
5.2 Approximations to Account for Hyperexponential Server	74

	Page
5.3 Sensitivity of Power to Variations in Predicted Values of Parameters	76
5.4 Design Choice of Buffer Size and Window	77
5.5 Optimal Time-out Factor	79
5.6 Independence of Outputs	80
5.7 Replacement of Effective Trans- mission Process by a Hyperexponen- tial Server	82
5.8 Stability of the Numerical Method	84
5.9 Some other Possible Areas for further Work	84
5.10 Summary	89
REFERENCES	105
APPENDIX A	113
APPENDIX B	116
APPENDIX C	119
APPENDIX D	126

CHAPTER 1

FLOW CONTROL IN PACKET-SWITCHED COMPUTER COMMUNICATIONS NETWORKS

1.1 WHAT IS FLOW CONTROL

In a dynamic sharing environment of packet switching close monitoring and control is necessary to ensure avoidance of undesirable effects like deadlocks and congestion. Routing [survey SCHW 80] and flow control [surveys KLEI 80b,POUZ 81] interact with each other in this task. At one extreme is loose coupling where flow control is activated only in case of a catastrophe. At the other end of the spectrum, is tight coupling where a packet is allowed to proceed only when the transit path is absolutely clear.

Congestion avoidance requires two decision making processes :

- i) Choice of time-epochs at which a packet is admitted into the network (or a part of the network, e.g., subnetwork, a node etc.). The time epochs of admission at various levels of the network are controlled by mechanisms at various protocol hierarchies. For example, admission to network is controlled by Network Access Protocol while admission to next node, by link level protocol.

- ii) Choice of path along which the packet reaches its destination with minimum delay subject to throughput constraints.

It may be interesting to note that while (i) is sufficient for congestion avoidance, a combination of (i) and (ii) is necessary for optimal performance, i.e., minimal delay subject to throughput constraints.

Routing accounts for decision process (ii). Decision process (i) is delegated to a set of mechanisms operated at various protocol hierarchies known as flow control. This also includes :

- i) error recovery mechanisms for data packets like re-transmissions or forward error correction (FEC).
(Physical links for digital communication are better modelled by channels with memory (e.g. to account for error bursts) than the memoryless BS channel. A survey of the models, various error control schemes and their suitability is done by Karal and Sastry [KANA 78]).
- ii) Adaptive bit sampling rates or wordlengths for voice. This is discussed in Section 1.8.

1.2 WHY PACKET SWITCHING ALLOWS BETTER CONTROL

Availability of hardware at cheaper prices has made possible a number of alternatives to traditional circuit

switching. These include enhanced circuit switching and packet switching. An excellent comparison is done by Harrington [HARR 80].

The nature of voice and interactive data traffic makes circuit switching inefficient. Voice traffic consists periods of talkspurt and silence. Interactive data consists of periods of activity and 'thinking time'.

Enhanced circuit switching dynamically allocates and deallocates circuits on sensing activity or silence respectively. Time Assigned Speech Interpolation (TASI) [BUIL 59] was started on long trunk routes for voice and is found more efficient if the number of sharable channels is sufficiently large.

The call connection is done by sensing beginning of a talkspurt. The fraction of traffic lost due to dynamic connection time or blocking due to all channels busy is known as cut-out fraction or freeze-out time. A cut-out fraction of 0.5 percent is acceptable for speech. Digital Speech Interpolation (DSI) is the digitized version. For interactive data, adaptive digital multiplexing, ADM, is used. This however does not perform as well as TAST because the error (cut-out fraction) tolerable is of the order of 10^{-7} for data. This factor also affects hybrid schemes as the fraction of data increases.

The only flow control measure in traditional CS systems is blocking. Voice traffic is better suited for this because of the delay constraints and no error recovery. Packetized voice is finding use. Weinstein [WEIN 78] showed that the cut out fraction in a TASI system is same as the fraction of packets dropped in a packet switched system with dropping if delay exceeds a certain threshold. Packet switching allows wider control than traditional circuit switching methods because

- i) Error recovery, advantages of adaptive routing are easily implemented in PS protocols.
- ii) Synchronization is crucial in CS systems.
- iii) Precedence and preemption are easily performed with PS protocols. In other words, PS systems can handle priorities better.
- iv) Packet switched systems can be designed for average load as opposed to peak load for CS systems.

Enhanced circuit switched systems are able to incorporate some of the advantages of packet switched systems. The distinction between PS and enhanced CS methods may be noted. If the number of outgoing channels is C , enhanced CS methods block subsequent customers if C are already active. PS methods allow more than C customers, with each customer facing greater delay. The mechanism of blocking can, of course, be introduced in a PS scheme as a call blocking protocol.

1.3 OBJECTIVES OF A FLOW CONTROL SCHEMA

The main functions of flow control (FC) in packet networks are [KLEIN 80b]:

- i) prevention of throughput degradation and loss of efficiency due to overload
- ii) deadlock avoidance
- iii) fair allocation of resources amongst competing users.
- iv) speed matching between network and attached users.

There are other objectives which are more related to a good economic balance between benefits and drawbacks of FC mechanisms [POUZ 81], namely

- a) minimize response-time
- b) maximize useful throughput
- c) minimize transmission cost
- d) keep receiver busy at all times
- e) minimize overheads.

The diversity of objectives indicates that there possibly cannot exist a single FC scheme. For example, (a) would be the most important factor for voice traffic while (b) and (c) may be more important for bulk data transfer. As another example, objective (a) would require smaller packet lengths which would not satisfy objective(e).

1.4 RESOURCES

The resources required to carry a flow of data can be

categorized into two classes [POUZ 81] :

- a) basic : buffers, transmission bandwidth, processor time
- b) incidental : name space, table entries, logical channels, etc.

Most studies consider buffers as the critical resource. In one case, CIGALE, mechanisms based on bandwidth were investigated. No study to the best of our knowledge has considered processor time a bottleneck. However, this aspect is expected to receive attention with faster communication lines e.g. T1 carrier (1.544 Mbits/sec) as compared to earlier 50 KBPS lines, coming into operation.

Incidental throttling mechanisms are implementation dependent. There can be finite number of call establishments depending on the capacity allocated to the name space table. A packet may be blocked if its allocated logical channel (e.g. channel no = packet no. MOD no. of channels) has not received the acknowledgement of previous packet even though the physical channel and receiver can accept packets. Also additional delay may be introduced by the overheads.

1.5 PERFORMANCE MEASURES

In a network there are two conflicting objectives :

- a) minimize delay
- b) maximize throughput

For example, throughput can be increased in a network by allowing more packets into it. But this means each packet encounters more waiting packets at each node. Hence, the delay is increased, and vice-versa. A performance measure that takes both effects into account is the throughput to delay ratio. This is termed 'power'. Bharath-Kumar and Jaffe [BHAR 81] have done extensive work in the properties of this function. One interesting aspect [JAFF 81] is that flow control power is non-decentralizable. That is, any method to maximize power must consider the whole state of the network.

This may be compared with the fact that there are efficient de-centralized routing algorithms.

Lam [LAM 78] introduced a factor known as bursty factor. He argued that 'burstiness' of traffic on a network is not only dependent on the nature of generation at source but also on the characteristics of communication links being used. The factor is the transmission rate along the line.

On a slow line, bursty data may be received more continuously whereas on a very fast line data generated with a high duty cycle can be received in a low duty cycle, and hence 'bursty'.

$$\text{Bursty factor} = \frac{\text{Specified message delay constraint}}{\text{Average interarrival time between messages}}$$

Bursty further is an upper bound of duty cycle. Traffic with low bursty factor is suited for PS. However, high bursty factor would not necessarily be better suited for CS.

1.6 LEVELS OF FLOW CONTROL IN DATA NETWORKS

Flow control can be exercised at various levels of protocol hierarchies [KLEIN 80b]. The requirements are :

- a) Implementation of FC mechanism, at a level, must be consistent with other protocol functions at the same level.
- b) Interactions between various levels must be consistent.
- c) There should be no duplication of functions at different levels.

At the lowest level is the physical level. This is responsible for the electrical connections - activations/deactivation. No flow control function is possible.

Above the physical level is the link level. This layer purports to transport packets reliably across the physical links. Some of the flow control functions at this protocol level may be retransmission of packets not received correctly due to physical error, packets dropped due to congestion at next node, stop the sender (e.g. Receiver Not Ready (RNR) in HDLC, SDLC). HDLC (ISO defined standards) SDLC (IBM's SNA), X.25 level 2 (subset of HDLC) are examples of link protocol implementations.

Above the link level we have the packet level protocol. This is used for establishing end-to-end user connections, through the network. This carries control information to route packets. The two methods of flow are :

- a) Virtual circuits (VC), and
- b) Datagrams.

In datagram method, packets are controlled individually. Thus two packets from the same source may arrive at a destination node via different routes. In VC method, packets are received at the destination in the sequence of input.

Datagram controls are based on :

- a) buffers, mostly , or
- b) bandwidth, e.g. CIGALE.

In VC method the control is stream wise. Source-destination pairs are identified. This method allows selective control on the various traffic classes. It allows one more degree of control by restricting the number of VC's. The controls can be :

- i) End-to-end - the inner sublayer (transit through intermediate nodes) is a datagram structure. Examples of these are in ARPANET, DATAPAC.

ii) Step-wise selective control is applied at node level as well. Examples are TYMNET, TRANSPAC, GMDNET.

There are schemes where both datagrams and VC methods are used. These are known as hybrid schemes. An example is Integrated voice and data networks. Transmission of voice packets takes place in VC mode while data packets in datagram mode.

Above the packet protocol is the end-to-end (ETE) protocol. The main objective of this is reliable transportation of single/multi packet messages from network entry to network exit nodes. Important functions of this protocol are reassembly of multipacket, messages at exit and regulation of input traffic using buffer allocation and windowing techniques. Some implementations do not have the ETE level in which case this function is delegated to higher level protocols.

The highest level of network protocols which carries FC functions is the Transport level. This provides for reliable delivery of packets on the 'virtual circuits' between 2 remote processes. The goal is to regulate the flow so as to make the most efficient use of network resources. 'Window' and 'Credit' schemes are generally used for this purpose.

1.7 FLOW CONTROL METHODS USED AT VARIOUS LEVELS OF PROTOCOLS

a) Hop Level

The main control techniques by observation of buffer sizes. Finer's control may be exercised by observation on the number of unacknowledged packets (window size).

The methods used are : (i) Channel Queue Length (CQL)

The classes of queues correspond to the various output channels. The restrictions placed would be the number of buffers each class can seize. At one extreme is the completely partitioned (CP) scheme. This is dead lock-free but the utilization of resources is poor. The other extreme is completely shared (CS) scheme. The utilization is increased but this is prone to deadlocks and is unfair to input traffic of lower intensity. In practice a combination of the above methods SMXQ (maximum Q length), SMA (minimum allocation), SMXQA (maximum Q length with minimum allocation) are found to be good. Kamoun and Kleinrock [KAMO 80] have analyzed the performance of all the above schemes.

CQL avoids thruput degradation, unfairness, direct store-and-forward deadlocks.

However, indirect S/F deadlocks cannot be avoided by CQL. This happens for example in a ring topology where each node wants to forward to the next node but each node has its buffer full.

ii) Structured Buffer Pool (SBF)

The buffers are divided into classes. The class of buffers that a packet can seize depends on the number of hops it has made. Indirect S/F deadlocks are prevented. This can be proved by resource graphs [KLEIN 80 b]. GMD network uses this scheme [GIES 81].

iii) Virtual Circuits

Selective control is made on each traffic class. This is the same as step-wise VC control scheme discussed in Section 1.6.

b) End-to-End

This protocol is designed to work between source-destination pairs. The main objective is to protect the exit (receiver) node from congestion. An important by product is the prevention of global congestion, by reduction of the number of packets in transit in the network. This is based on end-to-end window scheme.

c) Network Access

The objective in this case is to reduce the transit delay at the cost of admission delays. A control on the total number of packets in transit in the network can reduce congestion.

The methods used are :

i) Isarithmic Scheme [DAVI 79]. Packets are allowed to proceed if they can secure a permit. The number of permits in the network is fixed. Isarithmic method requires an End-to-End scheme to maintain fairness in the distribution of the permits.

ii) Input Buffer Limits

Transit packets are given preference over entering packets. Also packets having travelled longer hops (closer to destination) are given preference over those which are fairly 'newer' in transit. Lam and Reiser [LAM 79] and Kamoun [KAMO 81] have studied these schemes. The number of buffers available to entering packets at gateway nodes, is restricted.

iii) Choke Packet Scheme

The network CIGALE uses this scheme. The emphasis is on the physical packet length rather than maximum packet length rather than maximum packet length in case of buffer schemes. This is done by observation of link loading. When the loading of a line sensed by a channel load limiter exceeds 0.7, a signalling packet, called choke packets is sent to sources generating excess traffic to thin link. If reduction of the loading does not take place, e.g. a source ignoring signal, a tripping mechanism automatically drops packets, on the link, causing congestion.

d) Transport Level

The objective is to maintain maximum efficiency amongst processes. One of the factors to be considered is the loading of network to convey changes. For example, a graphics protocol (see discussion of 3 high level protocols, [SPRO 78]) is discussed by Sprouille and Cohen. Each change of scene would cause transmission of a large number of bits if information on all pixels are to be transmitted. By dividing the scene into segments, only the segment where change has occurred needs to be transmitted.

1.8 FLOW CONTROL FOR VOICE AND INTEGRATED VOICE AND DATA NETWORKS

The requirements for voice and data packets are somewhat complimentary. Data packets require high fidelity. This necessitates error recovery mechanisms. Delay is a secondary criteria to fidelity. Voice packets on the other hand, require a guaranteed delay, for continuity of conversations. Fidelity is not such a crucial criteria as loss upto 0.5 percent or slight distortions can be tolerated. The error recovery protocol is not activated for voice packets. Furthermore, to ensure guaranteed delay, voice packets are dropped if the waiting time at any node exceeds a threshold, chosen to satisfy the constraint of acceptable loss as well. Voice packets are generally sent in VC mode

while data packets can be sent in datagram or VC mode.

Flow control of voice also involves applications of efficient speech coding techniques [GOLD 77]. Voice control methods have to be more adaptive. Voice controls are incorporated from end-to-end level of protocol hierarchy. As the network gets loaded, the bit sampling rate is reduced within acceptable limits. Bially et al [BIAL 80] presented an end-to-end adaptive control scheme based on embedded speech coding. Each frame of 20 ms is divided into packets of different priorities. Depending on the state of network, link management has to decide the cut off priorities. Packets below this priority level are dropped. This method can be exercised to a point where the dropping ensures reasonable fidelity. This is advantageous over other schemes because of the ability of the link management to effectively reduce the bit rate. Other methods, involving end-to-end adaptive control suffer from the delay required to signal the end user.

The delay restrictions and suppression of error recovery, made voice suited for circuit switching in hybrid PS and CS networks. Frame multiplexing structure [WEIN 80] is used. A frame of duration 10 ms is chosen in view of delay requirements for voice. This is divided into two parts :

- a) a fixed number of slots of fixed duration for voice transmitted in synchronous mode with circuit switching
- b) the remaining portion available for data in asynchronous mode.

Fixed boundary case, when data packets cannot use vacant voice slots, is same as separate flow control schemes for voice, data traffic. When data is allowed to use vacant voice slots, the performance is better. This is the movable boundary case.

TASI, DSI, packet voice control with dropping suffer from the problems of cut-out fractions. Adaptive bit sampling introduces distortions. Another control scheme for voice is Predicted Wordlength Assignment [GERH 82]. In this method the sampling rate remains constant. However, the number of bits per sample or wordlength, is adaptively varied. This method avoids cut-out fractions and distortions but introduces noise.

1.9 FLOW CONTROL IN INTERCONNECTED NETWORKS

Flow control in network interconnection, under virtual circuit mode of packet transmission (it may be pointed out the X.25 allows packet forward in VC mode) has been studied by Matsuto and Mori [MATS 81]. This is based on the virtual reduction of calls.

Edge [EDGE 79] studied the End point approach, which uses a standard internet end-to-end protocol and the hop-by-hop approach in which inter net calls consist of a sequence of local virtual calls across each intermediate net.

Bennett [BENN 79] studied the problem of using trans-network architectures to support transnetwork bulk data transfer.

1.10 FLOW CONTROL FOR REAL-TIME COMMUNICATIONS

Cohen [COHE 80] pointed out the problems to be faced in flow control for real time communications. For example, in the schemes mentioned till now, a new packet is not accepted if the buffers are full. However, if the information carried by the packet is, say the position of an aircraft under surveillance, it would be useful to retain this packet and drop the older packet. The newer packet would convey more information to the receiver, in spite of dropping of intermediate information.

He named this scheme 'Milk Rule' as opposed to the conventional 'Honey Rule' where the latest packets to try entry are discarded.

1.11 THE AIM OF THIS STUDY

The efficiency of flow control scheme at a given level of protocol hierarchy depends on the schemes below it, at the lowest level is the link level control scheme. Problems

like noise in physical communication links affect the performance at this level.

Voice control schemes generally exploit the inherent redundancy in speech signals by reducing the bit rate. The stringent delay requirements achieved by dropping of packets at a node if a certain threshold is exceeded, further ensure that subsequent packets are not affected in terms of delay.

Data packets, are more vulnerable to noise resulting in serious degradation of the network performance because of the additional traffic and delay introduced by the error recovery mechanisms. Improper choice of time out can cause redundant retransmissions. Improper choice of limit on unacknowledged packets in transit (window size) can lead to frequent blocking at next node and hence retransmissions, or lower throughput than what is possible.

Consideration of these facts led to the development of a queuing model, the main result of this thesis, to study the performance of a general link-level protocol for continuous transmission with error recovery and finite window size. This model is found to be very useful for making design choice of the various parameters.

Chapter 2 presents a summary of results and relevant applications that were motivating factors for the development of the model and the method of analysis. Chapter 3 presents

a step wise development of the model and a numerical procedure to evaluate the necessary parameters. Chapter 4 discusses the design of a simulation experiment, implemented using PASCAL on DEC-10, to study the accuracy of the model. Chapter 5 presents the results, observations and guidelines for further work. It also points out some modifications to the numerical procedure based on results derived in the Appendices, giving more accurate results for moderate traffic range.

CHAPTER 2

PERFORMANCE EVALUATION IN COMPUTER NETWORKS

2.1 INTRODUCTION

Computer network nodes are modelled as queuing systems where the arrival process is the arrival of packets and the service process is the processing of the packets in buffer switching, transmission etc. These processes are strictly not deterministic. The packet arrivals do not occur at regular intervals of time and the packet lengths, which govern the transmission time (service) are not always fixed. Queuing effects become important in such systems where either the arrival process or the service process or both have variations. For example, when arrival and service processes are identical, and have variations, queuing theory predicts a non-zero expected queue length, contrary to intuitive thinking. Results of queuing theory are used to predict the relevant measures like average delay at nodes, throughput, buffer overflow probability, etc.

Queuing theory was used by Erlang to study the performance of telephone systems. His famous Erlang B-formula, the probability of C customers in $M/M/C/C$ queuing system, gives the call blocking probability and is used extensively in telephony.

Kendall [KEND 51] pointed out the three basic factors in queuing systems :

- a) arrival process
- b) service process
- c) service discipline.

The shorthand for representation of queuing systems was introduced by Kendall [KEND 53].

There are two approaches to the analysis of queues :

- i) difference-differential equations of Erlang
- ii) integral equations of Lindley [LIND 52].

2.2 SIMPLE QUEUES

The simplest queuing system is the $M/M/1/\infty$ system. The arrival process is poisson and the service time density function is exponential. The poisson arrival process gives rise to exponential inter-arrival time density function. Observations show that the demands of a large independent population, e.g. telephone call demands, tends to follow poisson distribution.

The exponential density function is the only one that gives rise to memoryless condition,

$$P[\tau < t < \tau + \delta] = P[t < \delta]$$

Let the inter-arrival time density function be, $a(t) = \lambda e^{-\lambda t}, t > 0,$

and the service time density function be,

$$b(t) = \mu e^{-\mu t}, \quad t > 0$$

Then, the state of the system is completely specified by the number of customers in the system whose probability [KLEI 75],

$$p_n = \rho^n (1-\rho), \quad n = 0, 1, 2, \quad (2.2.1)$$

where $\rho = \lambda/\mu$.

The condition for stability is $\rho < 1$.

Computer Network nodes have finite buffers. If N is the maximum number of customers in service or waiting, then the system $M/M/1/N$ gives rise to the state probabilities [KLEI 75, SCHW 77]

$$p_n = \begin{cases} (1-\rho)\rho^n/(1-\rho^{N+1}), & \rho \neq 1 \\ 1/(N+1) & , \quad \rho = 1 \end{cases} \quad (2.2.2)$$

The ergodicity condition is $\rho < \infty$.

The probability that an arriving customer is turned away due to no waiting room is the probability that N customers are already waiting or in service at the system. This is given by

$$p_N = (1-\rho)\rho^N/(1-\rho^{N+1}) \quad (2.2.3)$$

The expected number in the system, L , is given by [GROS 74]

$$i) \quad L = \rho / (1 - \rho) \quad \text{for } M/M/1/\infty \text{ system}$$

$$ii) \quad L = \begin{cases} \frac{\rho [1 - (N+1)\rho^N + N\rho^{N+1}]}{(1-\rho)(1-\rho^{N+1})}, & \rho \neq 1 \\ N/2, & \rho = 1. \end{cases}$$

$$\text{for } M/M/1/N \text{ system.} \quad (2.2.4)$$

The waiting time in the system (including service) is given by Little's Formula [LITT 61, JEWE 67] :

$$i) \quad W = L/\lambda, \quad M/M/1/\infty$$

$$ii) \quad W = L/\lambda', \quad M/M/1/N \quad (2.2.5)$$

$$\text{where } \lambda' = \lambda (1 - p_N) \quad (2.2.6)$$

is the effective throughput.

Apart from finite capacity queues, variations of the $M/M/1$ system applicable to computer networks include :

- i) State dependent inter-arrival times and/or service times. For example, the probability of arrival of a talkspurt or length of a talkspurt (arrival of silence period) depends on the number of active talkers [WEIN 78]. The analysis of W-R retrieval model for telephone systems [FRED 79] incorporates the effect of a constant retrieval probability by blocked customers by a state dependent service rate, by taking the marginal probabilities.

- ii) Queues with impatience applicable, e.g., to voice packets which are dropped after a certain delay threshold at a node [WEIN 78].
- iii) Bulk arrival or service, e.g. the arrival of acknowledgements under IBM's SNA virtual pacing scheme can be modelled as a bulk arrival process [SCHW 82].
- iv) Queues with finite population. Physical computer networks or systems cater to a finite population. However, if this is sufficiently large the finite population constraint, modelled via state dependent methods, is dropped. An application of finite population model is finding the number of interactive terminals attachable to a system beyond which the response is greatly deteriorated [KLEI 76]. This is known as the saturation number, M^* , given by, $M^* = 1 + \frac{\mu}{\lambda}$.

Analysis of these cases of the M/M/1 model is available in most books on queuing systems, e.g. [KLEI 75].

2.3 QUEUES WITH INTERMEDIATE COMPLEXITY

The queuing system becomes more complex when one of the processes is not Markovian. These are the M/G/1 queue where the service time can take a general distribution and GI/M/1 queue where the inter-arrival time distribution can be arbitrary.

2.3.1 M/G/1 Queue

The analysis of the M/G/1 queue uses the embedded Markov process approach of Kendall [KEND 51, KLEI 75]. The state of the system must now include the service already expended on the customer in service. For the M/M/1 queue the incremental service time required for completion for the customer in service is independent of the service received due to the memoryless property. The state vector $[N(t), X(t)]$ where $N(t)$ = number in system, $X(t)$ = service time spent on customer in service, is now a Markovian process. Kendall's ingenuity [KEND 53] was to consider the embedded Markov process, the state of the system at departure epochs.

The state equations can now be written as,

$$p_j = \sum_{k=0}^{j+1} p_k d_{j+1-k} \quad (2.3.1.1)$$

where d_k = probability of k arrivals since last departure. If $b(t)$ is the density function of the service time and λ , the parameter of the poisson arrival process, then,

$$d_k = \int_0^{\infty} \frac{e^{-\lambda t} (\lambda t)^k}{k!} b(t) dt \quad (2.3.1.2)$$

The parameters of interest can be found by transform methods [KLEI 75]. The z-transform of the system queue size is given by,

$$Q(z) = B^*(\lambda - \lambda z) \frac{(1-\rho)(1-z)}{B^*(\lambda - \lambda z) - z} \quad (2.3.1.3)$$

where $B^*(s)$ = Laplace Transform of service time pdf and λ is the parameter of the poisson arrival process. This is known as the Pollaczek-Khintchine transform equation. Taha [TAHA 71] has given the results for $M/E_m/1$ and $M/D/1$ systems.

The Laplace transform of the waiting time in queue is given by

$$W^*(s) = \frac{s(1-\rho)}{s - \lambda + B^*(s)} \quad (2.3.1.4a)$$

The overall system waiting time is given by $S^*(s) = W^*(s)B^*(s)$ (2.3.1.4b)

Finch [FINC 58] analyzed the $M/G/1$ system with finite waiting room of size, N . He showed that if $(1-x)/(d(x)-x)$ can be expanded as a power series in some region, then the generating function for state probabilities of interest is given by

$$P(x) = \frac{P_0(1-x)}{d(x) - x} \quad (2.3.1.5)$$

where $P(x) = P_0 + P_1x + P_2x^2 + \dots + P_Nx^{N-1} + P_{N+1}x^N$,

P_n = probability that number in system at departure epoch = n ,

and

$$d(x) = \sum_{k=0}^{\infty} d_k x^k$$

where d_k is given by eqn. (2.3.1.2), P_0 is determined to ensure that the sum of the probabilities is equal to 1.

Courtois [COUR 80] analyzed the M/G/1 finite queuing system in a different manner. The open M/G/1/N queue is replaced by a closed two stage cyclic queuing network with N cycling customers with exponential server (arrival process) at first stage followed by a general server at the second stage.

Considerable work in computer systems, e.g. scheduling, has been done under M/G/1 consideration. Multiprogramming, time sharing systems are advantageous when the coefficient of variation of service time is greater than 1 [KLEI 76]. The analogy to networks is that if the coefficient of variation of message lengths is greater than 1 packetization gives better performance.

A stop-and-wait protocol for link level was analyzed by Fayolle et al [FAYO 78] under M/G/1 assumption. Their objective was to predict optimal time-out periods to :

- i) minimize delay
- ii) maximize throughput
- iii) minimize buffer overflow probability with finite buffers.

Kuehn [KUEH 81] analyzed the performance of two different ARQ-protocols for half duplex transmission between a primary station and an arbitrary number of polled secondary

stations. He analyzed them under cyclic queuing models of the type $M^{[x]}/G/1$ with poisson batch arrivals and general service time processes.

2.3.2 GI/M/1 Queue

Let the arrival process have density function $a(t)$ and service time density function be exponential with parameter μ .

The observation epochs are now the arrival instants.

We have now, the state probabilities,

$$p_j = \sum_{k=0}^{\infty} p_{j+k-1} d_k \quad (2.3.2.1)$$

where d_k , the probability of k customers services since last arrival, is given by,

$$d_k = \int_0^{\infty} \frac{e^{-\mu t} (\mu t)^k}{k!} a(t) dt \quad (2.3.2.2)$$

The infinite summation in equation (2.3.2.1) as opposed to finite summation for the $M/G/1$ case (eqn. (2.3.1.1)) leads to a simple results for the state probabilities [TAHA 71, KLEI 75],

$$p_j = (1-\sigma)\sigma^j \quad (2.3.2.3)$$

where, σ is the solution in $(0,1)$ of

$$\sigma = A^*(\mu - \mu\sigma) \quad (2.3.2.4)$$

where $A^*(s)$ is the Laplace Transform of $a(t)$.

The waiting time for GI/M/1 queue follows exponential distribution.

An exponential server fed by the output of an M/G/1 system constitutes a GI/M/1 system.

2.3.3 The M/G/1 Conservation Law and Fairness Policies in Scheduling in Computer Networks

In computer networks, priorities have to be assigned to traffic classes. It is useful to find a way for evaluation of such systems. Kleinrock's famous M/G/1 conservation law states [KLEI 76] that for any non-preemptive work conserving queuing discipline,

$$\sum_{p=1}^P \rho_p W_p = \begin{cases} \frac{\rho W_0}{1-\rho} & , \rho < 1 \\ \infty & , \rho \geq 1 \end{cases} \quad (2.3.3.1)$$

where ρ_p is the traffic intensity of the pth class and

$\sum_{p=1}^P \rho_p = \rho$. W_p is the mean response time for the pth class

and W_0 = mean response time (service + waiting) using the Pollaczek-Khintchine formula.

The conservation theorem allows the network designer to reallocated the delays among classes without significantly affecting the overall mean-end-to-end delay.

Wong et al [WONG 82] applied this to the study of fairness. They defined a generalized fairness measure,

$$F = \frac{1}{T^2} \sum_{p=1}^{P'} \frac{\gamma_p}{\gamma} (T_p - T_p^t)^2 \quad (2.3.3.2)$$

where, T = mean overall delay

T_p = mean delay for p th class

T_p^t = target delay for p th class

γ_p, γ are throughput for p th class, overall traffic respectively.

$$\text{Let } T_p^t = \alpha_p K \quad (2.3.3.3)$$

where α_p is a parameter indicating the priority of the p th class and K , a constant that can be found by application of the conservation law,

$$\sum_{p=1}^P \frac{\gamma_p}{\gamma} T_p^t = T \quad (2.3.3.4)$$

Therefore,

$$T_p^t = \frac{\gamma \alpha_p T}{\sum_{p=1}^P \gamma_p \alpha_p} \quad (2.3.3.5)$$

The various priority schemes considered were :

- i) uniform delay fairness : $\alpha_p = 1$ for all p .
- ii) Hop dependent fairness : This case is applicable for location independent tariffs (e.g. Telenet). One might want the delay of a class to be proportional to the number of hops on the path used by that class.

This is done by setting

$$\alpha_p = \pi_p, \text{ the number of hops.}$$

iii) Traffic dependent fairness : Higher priority for class with higher traffic volume is allocated.

$$\alpha_p = 1/\gamma_p.$$

FCFS scheduling policy gives approximately the same fairness measure as the optimal policy for the hop dependent case. For the other two cases, FCFS performs poorly.

2.4 COMPLEX QUEUING SYSTEMS

The most complex queuing system is GI/G/1 system where the arrivals and service times can take arbitrary distributions. Use of GI/G/1 methods would allow prediction of upper bounds on delays in computer networks.

The method of analysis for GI/G/1 queue uses a random variable,

$$u_n = x_n - t_{n+1} \quad (2.4.1)$$

where x_n = service time for nth customer, and

t_{n+1} = inter-arrival time for the (n+1)th customer.

If $A^*(s)$ and $B^*(s)$ are the Laplace Transforms of inter-arrival time and service time density functions, then the Laplace Transform of the density function of the random variable, u , is given by [KLEI 76]

$$C(s) = A^*(-s) B^*(s) \quad (2.4.2)$$

The distribution of the waiting time is given by Lindley's Integral equation [LIND 52, KLEI 76],

$$W(y) = \begin{cases} \int_{-\infty}^y W(y-u) dC(u), & y \geq 0 \\ 0, & y < 0 \end{cases} \quad (2.4.3)$$

Kingman [KING 62, KLEI 76] showed that the limiting waiting time distribution under heavy traffic (traffic intensity close to 1 but strictly less than 1 to maintain stability) is exponential with mean

$$(\sigma_a^2 + \sigma_b^2)/2(1-\rho) \bar{t} \quad (2.4.4)$$

where,

σ_a, σ_b = standard deviation of inter-arrival, service times respectively

ρ = traffic intensity

\bar{t} = mean inter-arrival time.

Recent developments in GI/G/1 queues include :

- 1) analysis by discrete fourier transform by Ackroyd [ACKR 80]
- 2) approximation of the waiting time distribution, $W(x)$ in the mean and probability of delay, $P_d = 1-W(0)$, by exponential distribution of the form,

$$W_a(x) = 1 - Ce^{-ax}, \text{ by Fredericks [FRED 82].}$$

2.5 APPROXIMATION METHODS

Fluid approximation (approximation in mean) and diffusion approximation (approximation in mean and variance) are useful in performance evaluation [KLEI 76, KOBA 74a,b] especially since the exact methods are difficult for complex systems. These methods are applicable under moderate to heavy traffic. This is the range where the effectiveness of flow control schemes lie.

The above methods replace the discrete stochastic process by a continuous process. The basis is as follows :

If $\alpha(t)$ = number of arrivals in $(0, t)$, and

$\delta(t)$ = number of departures in $(0, t)$,

then, if $\alpha(t)$ is large compared to unity,

$$\lim_{t \rightarrow \infty} \frac{\alpha(t) - \overline{\alpha(t)}}{\overline{\alpha(t)}} = 0 \quad (2.5.1)$$

The same holds for $\delta(t)$, $\overline{\delta(t)}$. Thus, the discrete stochastic processes $\alpha(t)$ and $\delta(t)$ are replaced by the continuous deterministic processes $\overline{\alpha(t)}$ and $\overline{\delta(t)}$ for fluid approximation. The number in the system,

$$N(t) = \overline{\alpha(t)} - \overline{\delta(t)} \quad (2.5.2)$$

The diffusion approximation method replaces the discrete stochastic processes $\alpha(t)$, $\delta(t)$ by continuous Gaussian distributions by accounting for the variations as well [KLEI 76].

2.6 OUTPUT PROCESSES AND NETWORK OF QUEUES

Output processes are important because not all system parameters are observable and a measure of performance is throughput [PACK 78]. Output processes are also important when extension to network of queues is considered. Burke [BURK 56] showed that the output of a $M/M/1$ system is poisson. Finch [FINC 59] pointed out that for the $M/G/1$ system the interdeparture times are independent only if the distribution of service time is exponential. For finite waiting room cases, even with exponential server the independence is not maintained. Daley [DALE 68] showed that the outputs of $GI/M/1$ queue are independent if arrival process is poisson. Mirasol [MIRA 63] showed that the output of a $M/G/\infty$ queue is poisson. The engineering implication is that if the waiting time at a node is sufficiently small (effect of a server always available), poisson output may be assumed. Recent work on output processes include a survey by Burke [BURK 72] and some results on $M/D/s$ queue [PACK 78]. Burke [BURK 72] pointed out that the output of the $M/M/1/N$ queue, when the dropped customers are also included, is poisson.

These factors restrict closed form solutions to network of queues to limited cases.

Jackson [JACK 57] solved the problem for a network of $M/M/c/\infty$ queues. He arrived at the famous result known as Jackson's theorem. The joint probability of n_1 customers at service station 1, ..., n_i customers at service station i etc. is the same as the product of the probabilities of the state at each node, independently. Or,

$$p(n_1, n_2, \dots, n_m) = \prod_{j=1}^M p_j(n_j) \quad (2.6.1)$$

Burke [BURK 76] proved a fact conjectured earlier. The output of a $M/M/1$ node with feedback is not poisson, in spite of the fact that the queue size, at each node of the Jackson network, is as if it were fed by independent poisson input. Kobayashi and Reiser [KOBA 75] pointed out that the factor of interest is actually the mean work demand brought by a job, and not how this work is divided into individual visits to a node by routing.

In computer networks the assumptions of Jackson's theorem are not satisfied because [KLEI 76] :

- i) service times are proportional to message lengths and can be anything but independent
- ii) arrival times are conditioned on service times at nodes and thus cannot be independent.

If, however, the number of input channels to the node as well as the number of outgoing channels are sufficiently large then the dependencies reduce. It is also known that

requests from a set of large number of independent sources to a pool of shared resources form a poisson process. These factors led Kleinrock to formulate the famous Independence Assumption for Computer Networks. The basis for service times at different nodes being exponentially distributed independent random variables was thoroughly tested out by simulation.

In a packet network of fixed size packets, it would appear that service times would be constant. However, considering overall service factors like processing time at next node, loss, delay in receipt of acknowledgement etc., the exponential distribution assumption is a fairly good one.

Gordon and Newell [GORD 67a] showed that under certain conditions closed queuing networks lead to product form solutions. They also considered the finite queue case [GORD 68b]. Using the results of [GORD 67a], Buzen [BUZE 73] presented algorithms for efficient computation. Kuehn [KUEH 79] used approximations by decomposition method to analyze network of queues under GI/G/1 assumptions. Kaufman et al [KAUF 81] used sparse matrix methods for analysis using Markov state space model. More details can be incorporated, e.g., deadlock probability in suppressing acknowledgement. However, the complexity of solution increases as the number of states increase.

The generalization of Jackson's theorem came from Baskett et al. [BASK 75]. They showed that under the following conditions :

- i) server exponential - discipline FCFS. Service time distribution is identical for all classes of customers.
 - ii) Processor sharing, server having rational Laplace Transform. Each class may have distinct service time.
 - iii) The number of servers is greater than the maximum number of customers. Each class of customer may have distinct service time, having rational Laplace Transform.
 - iv) Discipline LIFO, servers having rational Laplace Transform. Each class may have distinct service time.
- closed form solutions result. These systems are known as BCMP queues.

Cox [COX 55] showed that any distribution having rational Laplace Transform can be represented by stages of exponential servers has shown in Fig. 2.6.1. This includes feedback systems. Another representation can be given in terms of parallel exponential stages.

The conventional approach to the solution of queuing networks makes use of global balance equations. Whittle [WHIT 68] introduced a simplification known as independent balance equations. These equate the rate of flow into a

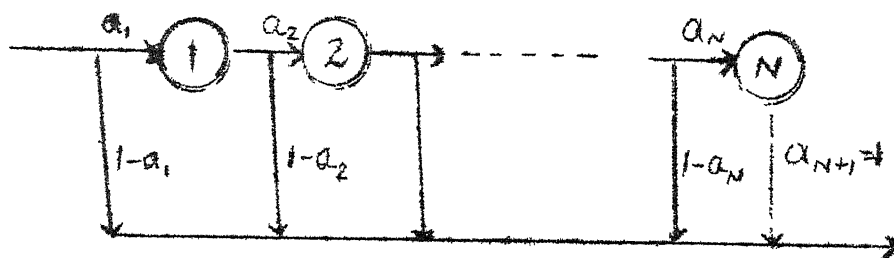


Fig. 2.6.1

state by a customer entering a stage of service to the rate of flow out of that state due to a customer leaving that state of service. Each global balance equation is the sum of independent balance equations.

Discrete time queueing systems and their networks have been studied [BHAR 80, HSU 76]. Examples of these include modelling of processor time (service) which is a multiple of a fixed quantum or allocation of slots for transmission which are multiples of a fixed quantum. Inter-arrival times may also be discrete, in terms of multiples of a fixed quantum. Unlike the Jackson's analog network, a network in which packets arriving in independent Bernoulli stream, are being serviced by a resource with geometric distribution, the independence property of outputs is not maintained.

2.7 APPLICATIONS OF NETWORK OF QUEUES

2.7.1 Some Open Queuing Models

Schweitzer and Lam [SCHWE 76] used the results of BCMP queues to calculate the blocking probability at a node with many outgoing channels under sharable buffer scheme, under markovian assumptions for arrival and service times.

They gave a closed form solution to their model of the form, $P(n_0, \underline{N}, \underline{M}, \underline{L})$, where n_0 = number at CPU (input) queue, $\underline{N} = (n_1, \dots, n_m)$, where n_i is the number at service center i for transmissions. $\underline{M} = (m_1, \dots, m_m)$ where m_i is the number transmitted and waiting for acknowledgement, and will be acknowledge at service station i . $\underline{L} = (l_1, \dots, l_m)$ where l_i is the number transmitted but will not be acknowledge and will be fed back to end of transmission queue after time out.

It is assumed that once a packet is transmitted there is always a server available to handle the acknowledgement process (This allows application of assumption 3 of BCMP queues).

Irland et al [IRLA 77] gave a numerical method of analysis for a network of nodes in tandem. The influence of finite buffering at each node, on blocking input packets and their subsequent retransmissions was considered.

Their results show that host-retransmission schemes and node retransmission schemes result in approximately the same delay, under the assumptions made.

2.7.2 Some Closed Queuing Network Models

Window schemes are easily modelled in terms of closed queuing networks. A finite number of customers (window size), can be circulating. This is because at one extreme W packets can be traversing and at the other extreme the W acknowledgements can be traversing back, and

a combination summing to W where W is the window size, is possible.

Reiser [REIS 79] modelled a network using window control. The open queuing network was converted into a closed multichain queuing network. For example, chain could be due to end-to-end window. Another chain could be due to the link-level window. Mean value analysis of Reiser and Lavenburg [REIS 80] and convolution method [REIS 81a] are used to analyze these networks.

Schwartz [SCHW 82] showed that the optimal end-to-end window size for a L -hop path is, $W_0 = L$. This assumption assumed identical servers. This allowed replacement by a single state dependent server by use of Norton's theorem [CHAN 75]. Also, the analysis was performed under saturation conditions (traffic intensity = 1) on the basis that this is the region where effect of flow control is most prominent.

Reiser [REIS 81b] introduced the effect of admission delays in a VC route caused by blocking due to lack of permit (window full) and showed that the optimal window for the end-to-end scheme is, now,

$$k_o = 2L$$

End-to-end data flow analysis based on closed queuing networks with application to HMINET was done by Baum et al [BAUM 79]. They found that the size of buffer space needed for transaction buffering and throughput are dependent on workload of the local computer. They thus, recommended a variation of the level of multiprogramming as a possible traffic control measure.

2.8 ZERO INSERTION AND ITS INFLUENCE ON PACKET LENGTHS FOR HDLC PROTOCOLS

The general format of an HDLC information (I) frame is shown in Fig. 2.8.1.

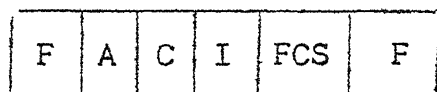


Fig. 2.8.1

It consists of

- i) a one byte flag (F) ,
- ii) a one byte address field (A),
- iii) a one byte control field (C)

- iv) arbitrary bytes of information (I),
- v) a 2 byte frame check sequence (FCS), and
- vi) the one byte flag (F).

The other formats e.g. supervisor are of fixed length. It is the I frames which have major influence in the performance.

The one byte flag (F) consists of a single '0' followed by 6 ones and terminated by a '0' - 01111110. This is used for synchronization. To avoid false synchronization due to this sequence in other fields, a 0 is added after each sequence of 5 consecutive ones. Ma [MA 82] derived the results for efficiency under general conditions where, a zero is added after n consecutive one's, using a Markov model. The optimal value of n, is,

$$n_{\text{opt}} = \log_2 \left\{ 2 + \frac{L \ln 2}{4} + \sqrt{\left(2 + \frac{L \ln 2}{4}\right)^2 - 4} \right\} - 1 \quad (2.8.1)$$

where L is the frame length.

The efficiency for general n and length L,

$$\text{EFF}(n, L) = \frac{L}{\text{FL}(n, L)} \quad (2.8.2)$$

$$\text{FL} = L \cdot \frac{2^{n+1} - 1}{2^{n+1} - 2} + 2(n+3) \quad (2.8.3)$$

where FL = resultant length due to zero insertions.

Ma's results show :

- i) under noiseless lines $EFF(5,L)$ is within 1-2 percent of $EFF(n_{opt}, L)$ hence the HDLC choice is good
- ii) under noisy lines, $EFF(5,L)$ is not as good as $EFF(n_{opt},L)$, beyond a limit for L . This is because greater the frame (packet) length the larger the number of zero insertions. One solution is to restrict the packet length to the acceptable value of L , and use the present HDLC format. Ma, on the other hand suggests a flag field of 2 bytes and FCS of 4 bytes. This is to make the flag field consisting of a '0', 14 consecutive 1's a '0'. Now the zero insertion takes place after every 13 consecutive 1's in other fields. This is more efficient and $EFF(13,L)$ is found to be closer to $EFF(n_{opt},L)$ for L exceeding certain limit.

2.9 DYNAMIC FLOW CONTROL

Kermani and Kleinrock [KERM 80] studied dynamic flow control. The parameters like time out, window size under end-to-end control are chosen by a Markovian decision model based on previous observations. This is found to give better performance than static flow control [KLEI 80a].

2.10 FLOW CONTROL IN INTEGRATED VOICE AND DATA NETWORKS

Analysis of flow control in integrated voice, data

networks are also being carried out. Fischer and Harris [FISC 76] presented a model for analysis of the integrated multiplex structure where each frame is subdivided into fixed slots for voice in CS mode and the remaining for data in PS mode.

Weinstein et al [WEIN 80] studied by simulation flow control of voice and data. Data packet control is effected by buffer limitation while voice packets are controlled by adaptive bit sampling. Smoothing algorithms [JAYA 76] can be used to handle transmission errors for voice.

Ross and Mowaffi [ROSS 82] presented an M/G/1 model for study of the parameters of data packets in a hybrid scheme. The server is modelled as one with rest periods (to service the voice slots in CS mode). The rest period is constant for a fixed boundary case.

2.11 PROPOSED QUEUING MODEL

Open queuing models are useful for predicting throughput, delay and buffer overflow probabilities, but do not account for blocking due to windowing effect. Closed queuing models, on the other hand, do handle window effect. However, the buffer overflow probability and retransmissions are not handled by these models.

This led to the development of a model for link level flow control incorporating buffer overflow, retransmission

after time out and windowing, presented in Chapter 3. Open queuing models of a node are useful for extension to network of queues and finding suitable approximation using BCMP results. Using the ideas of Gordon and Newell [GORD 67a], transformation of a closed queuing system into open queuing system with state dependent input buffer allocation is done. A numerical method, using ideas of Irland et al [IRLA 77] and Hillier et al [HILL 67], with considerable simplifications is formulated. It is found that for most acceptable values of buffer size and window in networks the input buffer allocation can be assumed independent and hence the open model is found quite accurate.

An easy to evaluate approximation is introduced to account for the under-estimation of delay with $M/M/1$ assumption under moderate traffic. Use of the exact methods, e.g. $M/G/1/K$ would be far more complex.

CHAPTER 3

A QUEUING MODEL FOR SOME FLOW CONTROL SCHEMES

3.1 INTRODUCTION:

In this chapter, we develop a queuing model for link - level flow control schemes incorporating retransmissions, time out and window. This model is depicted in Fig. 3.1.

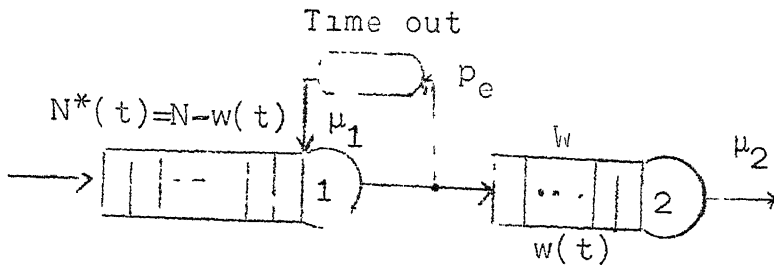


Fig. 3.1

Assumptions made in the model are:

Arrivals to the node are poisson with parameter λ . The packet lengths are distributed according to exponential distribution. The time for transmission proportional to packet length is exponentially distributed with parameter μ_1 . This is represented by the server at stage 1.

At most N packets can be presented at the node. This corresponds to physical buffer of size, N . A transmitted packet continues to occupy the buffer till receipt of acknowledgement.

The buffer management scheme tags this buffer to the list of unacknowledged packets. Each packet is acknowledged. The acknowledgements are in order of transmission of packets. The process of waiting for acknowledgement after transmission is represented by stage 2. The queue, here, represents the fact that the acknowledgement for a packet is sent after the acknowledgement for the previous packets. The size of this queue (maximum number waiting or in service) is W , the window size.

The acknowledgement intervals (time between two acknowledgements) are assumed exponentially distributed, with parameter μ_2 , to account for the various factors like delay in processing of packet at next node for detection of transmission errors and despatching acknowledgement, piggybacking of acknowledgements on packets in reverse direction, propagation delays etc. It may be noted that the transmission time for a packet is the elapsed time between sending the first and the last bit of the packet.

The buffer management scheme explained earlier, leads to an arriving packet at time t , to the physical node finding the buffer availability for packets at the transmission process (stage 1),

$$N^*(t) = N - w(t) \quad (3,1,1)$$

where $w(t)$ is the occupancy by packets waiting acknowledgement at time t . Thus, the inherently closed queuing system with W circulating customers, is converted to an open queuing system with the buffer availability $N^*(t)$ dependent on occupancy by unacknowledged packets $w(t)$. The analysis of this system would be complex. We shall simplify further, shortly.

If all $N^*(t)$ buffers are full (corresponding to all N buffers and full) an incoming packet is dropped. Let the probability of this event be p_1 .

Consideration of physical buffer sizes, window and expected occupancy by unacknowledged packets led us to make a simplification. The average buffer occupancy by unacknowledged packets is small as compared to N . Thus, we have the approximation, from (Eqn. (3.1.1)),

$$N^*(t) = N \quad (3.1.2)$$

We have, therefore, modelled a completely shared buffer management scheme as a completely partitioned scheme with N buffers available to incoming packets and W for unacknowledged packets. Further more the complex system is now modelled as a tandem 2-stage queuing system, with independent, finite buffers. The average occupancy by unacknowledged packets is of the order of 2 to 3 (this is confirmed by results of simulation), when the acknowledgements arrive at the same rate as transmission of

packets. This is because the finite buffering (stage 1) blocks input traffic to physical node under heavy traffic conditions. The acknowledgement process, therefore, caters to an already regulated traffic. The acknowledgement process is much faster than transmission process under normal conditions. This is because the acknowledgement packets are much smaller, and due to the effects of acknowledging upto a certain number of packets in a single acknowledgement, piggybacking etc.

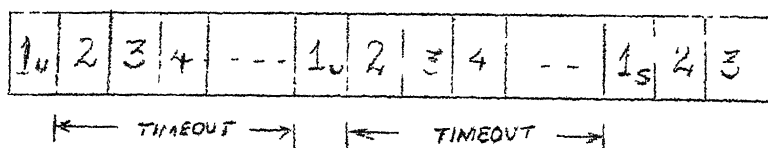
The above arguments would require careful review for satellite and long-distance terrestrial channels. The window size has to be much larger (e.g. in HDLC extended mode, $W = 128$) to allow continuous transmissions in presence of larger propagation delays (round-trip delays).

An accepted packet receives service at stage 1, which corresponds to the transmission process. The queue discipline is FCFS. With a probability, p_e , the packet has to be retransmitted. This may be because of erroneous delivery of packet or acknowledgement in transmission or blocking due to buffers full at next node. The retransmission takes place after a fixed duration, called time-out. This procedure repeats till successful transmission. We assume that time-out = 1. mean transmission time = $i \cdot \frac{1}{\mu_1}$ (3.1.3)

The Go-Back-N method of retransmission is used because of the sequencing assumptions. A packet will not be accepted out of sequence i at a node. In the actual flow control scheme, when a packet has to be retransmitted after the timer interrupt (time-out), the subsequent packets (at most $W-1$) are also retransmitted. If $i > W$ then the server at stage 1 is idle on the average for the duration,

Time out - time to transmit W packets, if an unsuccessful transmission takes place. In our model, we incorporate this effect by assuming that server 1 is busy during the entire time-out period and include this time in the service time of the packet requiring retransmission. The transmissions of other packets in-between are ignored. The constant time-out period is modelled as an Erlang- i stage with i exponential servers in series, with mean $\frac{1}{\mu_1}$, each.

With the above considerations the average service time of a packet is,



$$\begin{aligned} \bar{X}_{eff} = & (1-p_e)\bar{X} + p_e(1-p_e)(1+2)\bar{X} + p_e^2(1-p_e)(2i+3)\bar{X} + \dots \\ & + p_e^j(1-p_e)[j(i+1)+1]\bar{X} + \dots \end{aligned} \quad (3.1.4)$$

The model as formulated is quite complex and its analysis is not straight forward. To simplify the task we carry out the analysis in a step-by-step manner. We do this by making various simplifying assumptions which reduce the model to easily-solvable cases. These cases correspond to simpler versions of the generalized flow control scheme. Their analysis helps us in finally analyzing the general model.

The various versions considered by us are:

- a) Continuous transmission with infinite window
 - i) no error recovery
 - ii) retransmission after zero time out
 - iii) retransmission after finite time out
- b) Send-and-wait schemes
- c) Continuous transmission with finite window, no error recovery and, finally
- d) Continuous transmissions with finite window and retransmission after finite time-out (most general version).

The remaining sections of this chapter describe the analysis of these versions.

3.2 CONTINUOUS TRANSMISSION WITH INFINITE WINDOW:

Assuming an infinite window size implies that the probability p_2 of blocking due to second stage is zero. Since the second stage influences the first stage only through block blocking, we can eliminate the second stage. The system is now a single server with finite waiting room which is easy to analyze.

3.2.1 No Error Recovery:

This is equivalent to assuming that $p_e=0$, so there are no retransmissions. (There may be retransmissions due to acknowledgements delayed beyond time out. Analysing the effect of these redundant retransmissions is postponed to avoid complication). Hence the service time distribution is simply exponential with parameter, μ_1 , reducing the entire system to $M/M/1/N$ whose performance can easily be determined (Sec. 2.2).

3.2.2 Retransmission after Zero Timeout:

This corresponds to a hypothetical system in which the acknowledgement information is available as soon as it is transmitted. Note that the analysis of servers in series is done in time domain by convolution, which corresponds to product in L-T domain.

Referring to Fig. 3.1, with Time-out = 0, the Laplace Transform of the service time,

$$B_1(s) = (1-p_e) \left(\frac{\mu_1}{s+\mu_1} \right) + p_e(1-p_e)(1-p_e) \left(\frac{\mu_1}{s+\mu_1} \right)^2 + \dots + p_e^J(1-p_e) \left(\frac{\mu_1}{s+\mu_1} \right)^J + \dots \quad (3.2.2.1)$$

or

$$B_1(s) = \frac{(1-p_e) \left(\frac{\mu_1}{s+\mu_1} \right)}{1 - p_e \left(\frac{\mu_1}{s+\mu_1} \right)}$$

or

$$B_1(s) = \frac{\mu_1(1-p_e)}{s+\mu_1(1-p_e)} \quad (3.2.2.2)$$

Thus, it is as if the service rate is deteriorated to,

$\mu_{1 \text{ eff}} = \mu_1(1-p_e)$, the distribution remaining exponential. We can now apply the equations for an M/M/1/N system.

3.2.3 Retransmission after Finite Time-out:

Let the time-out = $i \cdot \frac{1}{\mu_1}$ as defined (Eqn. 3.1.3). With time out modelled as Erlang- i process. Each retransmission, one service time (exponential) in series with an Erlang- i (i exponential servers in series) timeout stage, is effectively on Erlang- $(i+1)$ stage. The L.T. of the

effective service time becomes (ref. Fig. 3.10 and eqn.3.14)

$$B(s) = (1-p_e) \left(\frac{\mu_1}{s+\mu_1} \right) + p_e(1-p_e) \left(\frac{\mu_1}{s+\mu_1} \right)^{i+2} + \\ + \dots + p_e^j(1-p) \left(\frac{\mu_1}{s+\mu_1} \right)^{j(i+1)+1} + \dots$$

or

$$B(s) = \frac{(1-p_e) (\mu_1/s+\mu_1)}{[1-p_e(\mu_1/s+\mu_1)^{i+1}]} \quad (3.2.3.1)$$

The effective service rate, is thus (ref. Appendix C),

$$\mu_{1eff} = - \frac{1}{B'(0)} = \frac{\mu_1}{1+(i+1)(p_e/1-p_e)} \quad (3.2.3.2)$$

We assume that the distribution may be assumed exponential. This approximation is valid because of the low value of p_e under normal operation (Appendix C, eqn. C.18). We can then use μ_{1eff} as the service rate in the M/M/1/N model to evaluate the performance.

3.3 SEND-AND-WAIT SCHEMES

In these schemes the acknowledgement for a packet must be received prior to transmission of the next packet. This effect is incorporated by assuming no waiting room in front of stage 2 in Fig. 3.1. The effective service consists of stage 1 server in series with stage 2 server ($p_2 = 1$), with no waiting room in front of stage 2 server.

Let $B_1(s)$ represent the Laplace transform of the pdf of the effective service time at stage 1. This corresponds to either of the cases discussed in Sec. 3.2. The L.T. of the pdf of overall service time, $B(s)$ is given by,

$$B(s) = B_1(s) \left(\frac{\mu_2}{s + \mu_2} \right) \quad (3.3.1)$$

where the second term on the right-hand side represents the L.T. of the pdf of service time at stage 2 (exponential).

The physical (successful) transmission from the node again corresponds to departure from stage 1. The service time at stage 2 is interpreted as delay for the next packet waiting for transmission.

When retransmission are not present ($p_e = 0$), the coefficient of variation of the effective service time at stage 1 is minimum ($=1$). The two exponential servers (stage 1 and stage 2) in series gives rise to the coefficient of variation of the overall service time less than 1. Therefore, modelling the system as M/M/1/N results in overprediction of the delay, nodal blocking probabilities etc. When retransmissions are present the hyperexponential nature of this stage brings the coefficient of variation of the overall service time closer to unity. M/M/1/N system analysis would therefore give better results.

3.4 CONTINUOUS TRANSMISSION WITH FINITE WINDOW AND NO ERROR RECOVERY

Referring to Fig. 3.1 this corresponds to the case when $p_e = 0$.

The effect of server 2 and the finite queues in front of server 1 and 2 prevents a closed form solution as in previous cases. A Numerical method is presented.

We start with an initial value of p_2 e.g. $p_2 = 0$. We incorporate the effect of blocking by server 2 by calculating the effective service rate of server at stage 1, as

$$\mu_{1\text{eff}} = \frac{\mu_1 \mu_2}{\mu_2 + p_2 \mu_1} \quad (\text{refer Appendix C, eqn. C.10}) \quad (3.4.1)$$

For small p_2 values (less than 0.2) exponential distribution for the effective service time at stage 1 may be assumed (ref Appendix C, eqn. C.14). The traffic intensity at stage 1 is calculated as $\rho_1 = \lambda / \mu_{1\text{eff}}$. Using the M/M/1/N equations we find p_0 , the probability of server at stage 1 idle (no packets at stage 1). The throughput from stage 1, λ_1 , is given by,

$$\lambda_1 = \mu_{1\text{eff}}(1 - p_0) \quad (3.4.2)$$

The second stage blocks output from first stage with prob. p_2 . For λ_1 to be the system throughput with blocking

probability, p_2 , the offered rate to the second stage, λ_2 , must be such that, $\lambda_1 = \lambda_2(1-p_2)$

or

$$\lambda_2 = \lambda_1 / (1-p_2) \quad (3.4.3)$$

Thus the traffic intensity at second stage ρ_2 may be calculated,

$$\rho_2 = \lambda_2 / \mu_2 \quad (3.4.4)$$

The new value of p_2 is calculated as the probability of W packets at stage 2,

or,

$$p_2 = \frac{\rho_2^W (1 - \rho)}{1 - \rho_2^{W+1}} \quad (3.4.5)$$

The equations (3.4.1) - (3.4.5) are iterated till convergence (which is possible if p_2 converges). The blocking probability at node 1, with the simplified assumption of Sec. 3.1, is the probability of N packets at stage 1.

or,

$$p_1 = \frac{\rho_1^N (1 - \rho)}{1 - \rho_1^{N+1}} = \rho_1^N p_0 \quad (3.4.6)$$

Thus we have :

Algorithm 3.1

begin

Initialize $p_2 = 0$

Until satisfied do

begin

$$\mu_{1\text{eff}} = \frac{\mu_2 \mu_1}{\mu_2 + p_2 \mu_1}$$

$$\rho_1 = \lambda / \mu_{1\text{eff}}$$

$$p_0 = (1 - \rho_1) / (1 - \rho_1^{N+1})$$

$$\lambda_1 = \mu_{1\text{eff}} (1 - p_0)$$

$$\lambda_2 = \lambda_1 / (1 - p_2)$$

$$\rho_2 = \lambda_2 / \mu_2$$

$$p_2 = \rho_2^W (1 - \rho_2) / (1 - \rho_2^{W+1})$$

end;

$$p_1 = \rho_1^N p_0$$

end;

3.5 CONTINUOUS TRANSMISSION WITH FINITE WINDOW AND RETRANSMISSION WITH FINITE TIME OUT

This is the most general case. We extend the algorithm developed in the previous section to account for the effective service at stage 1 with retransmission. This was dealt with in isolation, in Sec. 3.2.3. The effective service at stage 1 is hyperexponential. stage 2 provides a server in

series, with a probability, p_2 . This has the effect of reducing the variance. Thus we expect the overall service time to be closer to exponential than the cases in Secs. 3.2.3, 3.3 and 3.4.

We also introduce, at this stage, the probability of redundant retransmissions, p_r . A redundant retransmission occurs where the acknowledgement is delayed beyond time out. This has to be accounted for in some of the previous cases we had deliberately omitted to avoid confusion. Appendix D gives the method for calculating p_r (eqn. D.6). It is seen that p_r is a function of p_2 which depends on p_2 . Thus

algorithm 3.1 is to be modified to iterate over p_r and p_2 . We also append the calculation of the parameters of interest to complete the algorithm.

The effective service rate of server at stage 1 (not including blocking) is given using eqn. (3.2.3.2),

$$\mu_{1loss} = \frac{\mu_1}{1+(i+1)(p/1-p)} \quad (3.5.1)$$

where $p = p_e + p_r$.

p_e is constant but p_r changes in each iteration. Hence, μ_{1loss} changes in each iteration. The overall service rate at stage 1 including the blocking, is found from servers in series consideration (eqn. 3.4.1),

$$\mu_{1eff} = \frac{\mu_{1loss} \mu_2}{\mu_2 + p_2 \mu_{1loss}} \quad (3.5.2)$$

consideration of low p_e and p_2 values lead to exponential distribution approximation for the effective service time (eqn. C.16). Thus the general algorithm is :

Algorithm 3.2

begin

Initialize p_r , $p_2 = 0$;

While not satisfied do

begin

$$p = p_e + p_r$$

$$\mu_{1loss} = \frac{\mu_1}{1 + (i+1)(p/1-p)}$$

$$\mu_{1eff} = \frac{\mu_{1loss} \mu_2}{\mu_2 + p_2 \mu_{1loss}}$$

$$\rho_1 = \lambda / \mu_{1eff}$$

$$p_o = (1 - \rho_1) / (1 - \rho_1^{N+1})$$

$$\lambda_1 = \mu_{1eff} (1 - p_o)$$

$$\lambda_2 = \lambda_1 / (1 - p_2)$$

$$\rho_2 = \lambda_2 / \mu_2$$

$$p_2 = (1 - \rho) \rho_2^w / (1 - \rho_2^{w+1})$$

p_r is given by eqn. (D.6).

end;

$$p_1 = \rho_1^N p_0$$

$$\gamma = \lambda_1$$

$$Q_1 = \frac{\rho_1 [1 - (N+1) \rho_1^N + N \rho_1^{N+1}]}{(1 - \rho_1)(1 - \rho_1^{N+1})} \quad (\text{eqn. 2.2.4})$$

$$D = Q_1 / \gamma \quad (\text{eqn. 2.2.5})$$

$$P = \gamma / D$$

end;

where γ = throughput from physical node,

Q_1 = queue length at stage 1

D = delay faced by packet at physical node.

CHAPTER 4

DESIGN OF SIMULATION EXPERIMENT

4.1 RUNLENGTH CALCULATIONS

To validate the analytic model of Chapter 3, Monte Carlo simulation was used. The runlength was determined for 95 percent confidence at 5.0 percent tolerance of the mean blocked waiting times. A relative tolerance value, rather than a fixed value, was used because of the wide variations in waiting time with different arrival and service rates.

The observations were divided into blocks to reduce the autocorrelation. The mean of the waiting time for block was taken as the observation for runlength calculation purposes. The number of blocks was determined using the criteria derived from the Central Limit Theorem [DEO 79.Chapt. 7]. It was observed that block sizes of the order of 200 were necessary to ensure little correlation between the mean of waiting time per block.

4.2 IMPLEMENTATION OF SIMULATION MODEL IN PASCAL

The names of variable used in this chapter are based on the notations of Chapter 3. An array INPUTQUEUE, of size N (physical buffer size) stores information on packets waiting, being transmitted (in service) or waiting for acknowledgement. The information stored is the time at which a packet is to be

packet is to be appended to the table (This will be explained in the next few paragraphs).

A routine INSERTQUEUE accepts the array on which updating is to be done, the time of arrival the maximum number of entries as parameters. It updates the table by finding the position j , such that the last j packets have departed prior to the arrival. It calls a routine REMOVEQUEUE. This routine accepts as parameters the array on which updating is to be done and the number to be removed in a FIFO manner. This is done by moving the entries j places up starting from the $(j+1)$ st position. It returns the number currently in the queue. This is subsequently returned by INSERTQUEUE a calling sequence. INSERTQUEUE also returns the position where the information on the current arrival is to be placed. INSERTQUEUE signals via a boolean variable QSTATUS whether the packet can be accepted or not, due to buffer limitation.

The waiting time for transmission is calculated recursively. Assuming the current packet under observation is $(n+1)$ th packet, the waiting time,

$$W_{n+1} = \max(0, TC_n - a_{n+1}) \quad (4.2.1)$$

where TC_n corresponds to time of successful transmission of the previous packet and a_{n+1} the time of arrival of current packet.

The (n+1)th packet is ready for transmission at time,

$$\gamma_{n+1} = a_{n+1} + w_{n+1} \quad (4.2.2)$$

The 2nd table (array), ACKqueue, of size W stores the time of receipt of acknowledgement, (departure time from node) for packets waiting for acknowledgement at last update. This is scanned to check if transmission of the packet ready for transmission is to be blocked because of window constraints. The same two routines INSERTQUEUE and REMOVEQUEUE are used. The parameters passed now are the array ACKqueue, the time of arrival as γ_{n+1} and the maximum length as W. The queue length (Q_2) is returned.

If W packets are occupying ACKqueue, the time of transmission attempt of the packet ready for transmission is delayed by a factor δ_{n+1} accounting for the blocking. This is accounted in the waiting time, w_{n+1} , by incrementing it by δ_{n+1} .

Or,

$$w_{n+1} = w_{n+1} + \delta_{n+1} \quad (4.2.3)$$

where δ_{n+1} = time of receipt of acknowledgement of the first packet in ACKqueue - γ_{n+1} (4.2.4)

The first packet in ACKqueue is removed by calling procedure REMOVEQUEUE.

The time of start of transmission for the packet ready

is modified to,

$$\gamma_{n+1} = \gamma_{n+1} + \delta_{n+1} \quad (4.2.5)$$

The transmission process starts now. The time of transmission (service), S_{n+1} , is determined by generating a random variable from exponential distribution with parameter μ_1 . It may be noted that $\mu_1 = \mu C$ where $1/\mu$ is the mean packet length in bits and C , the transmission capacity of line bits per second.

The transmission process checks if a retransmission due to physical errors in transmission of packet, the acknowledgement or blocking due buffers being full at next node is necessary. Further more after a successful transmission and acceptance, several redundant retransmissions may take place because of the delay in receipt of the acknowledgement beyond time out period.

A procedure, RETRANSMIT, accounts for all of these effects. The time out factor, i , specified by the user, is used to calculate the constant time-out period,

$$\text{Time-out} = \frac{1}{\mu_1} \cdot i = \frac{i}{\mu_1} \quad (4.2.6)$$

The transmission of a packet, attempting transmission at γ_{n+1} , is completed at time,

$$TC_{n+1} = \gamma_{n+1} + S_{n+1} \quad (4.2.7)$$

A random variable is sampled from uniform distribution in $(0,1)$ to determine whether the transmission, acknowledgement contained errors or the packet was not accepted. This is determined by checking whether the sampled value is less than the probability of the above events, specified by user.

If the transmission is deemed unsuccessful, the next transmission is attempted at time,

$$Y_{n+1} = TC_{n+1} + \text{time out} \quad (4.2.8)$$

This process is continued till successful transmission.

The calculations on the time of receipt of acknowledgement is now done. This also includes the effects of redundant transmission.

The acknowledgement for the packets are assumed to be received in the same sequence as their transmission. The acknowledgement intervals are sampled from an exponential distribution with the parameter, μ_2 .

The time of receipt of acknowledgement,

$$D_{n+1}^{(2)} = \max(D_n^{(2)} + S_{n+1}^{(2)}, TC_{n+1} + S_{n+1}^{(2)}) \quad (4.2.9)$$

where

$D_n^{(2)}$ = time of receipt of acknowledgement for the previous (nth) packet,

$S_{n+1}^{(2)}$ = the acknowledgement interval for the (n+1)th packet.

If $D_{n+1}^{(2)}$ is greater than the next retry time for the packet, γ_{n+1} , another retransmission is performed. Transmission is complete at,

$$TC_{n+1} = \gamma_{n+1} + S_{n+1}$$

The next retry time,

$$\gamma_{n+1} = TC_{n+1} + \text{time out is calculated.}$$

If this is still less than $D_{n+1}^{(2)}$, another retransmission is attempted. This process continues till the next retry time is greater than $D_{n+1}^{(2)}$. These retransmission represent the redundant transmissions. $D_{n+1}^{(2)}$ is the entry stored in the tables, Inputqueue and ACKqueue for the current packet. All the acknowledgements received during redundant retransmissions are ignored for the purposes of the packet(n+1th) under observation.

For the next packet however, the recursive equation (4.2.9) assumes the time of receipt of acknowledgement of the previous packet as the last of the acknowledgements received. We could have made more realistic assumptions.

However, the effect of these retransmissions means delay for subsequent packets. The effect of the window, postpones the delay to the $(n+1+W)$ th packet. Since the statistical behaviour is of interest, incorporating the delay to the next packet, makes no difference.

4.3 RANDOM NUMBER SEQUENCES USED

Five different uniform random number sequences, borrowed from FORTRAN library, were used :

- a) RNDY1 - this is initially seeded by the user interactively. Samples from these random variables are used to seed the other sequences.
- b) RNDY2 - to generate interarrival times of packets from exponential distribution with parameter μ_1 .
- c) RNDY3 - to generate transmission time (service time) of packets from exponential distribution with parameter μ_1 .
- d) RNDY4 - to check for loss of packet in transmission, loss of acknowledgement or blocking at next node.
- e) RNDY5 - to generate the time to transmit the acknowledgement from adjacent node.
This is sampled from an exponential distribution with parameter μ_2 .

4.4 DIFFERENCES BETWEEN SIMULATION AND ANALYTIC MODEL

Simulation models must be kept as close to reality as possible. Assumptions in analytical models, however, have to consider the ease of mathematical tractability.

In the present study the points of similarity between the analytical and simulation models are :

- i) the interarrival times of packets are independently distributed random variables from exponential distribution,
- ii) the transmission times (based on packet lengths) are independently distributed random variables from exponential distribution.
- iii) the time for acknowledgements are independently distributed random variables. This is true even for retransmission. The last assumption in the simulation model is based on the observation that the acknowledgement is strictly not governed by the length of acknowledgement packet. Other factors like delay in processing at next node, transmission and propagation delays and piggybacking of acknowledgements are also involved.

The points of difference are :

- i) The simulation model uses a constant time out period while the analytical model assumes the time out to be a random variable sampled from a Erlang-i distribution,

where i is the time out factor.

- ii) During a retransmission, the originally sampled service time (time for transmission of packet) is used in the simulation model. This is because the physical packet length cannot change. The analytical model assumes an independently sampled value of service time (stage 1) for each retransmission.
- iii) Dynamic sharing of buffers between stage 1 and stage 2 is assumed in the simulation model. The analytical model assumes completely partitioned buffers where N is available from incoming traffic and W for packets waiting for acknowledgement. This is based on physical buffers size, N being sufficiently large as compared to the expected occupancy by unacknowledged packets, in network nodes.
- iv) In the simulation model, the actual blocking time is calculated for a packet that cannot be transmitted till the receipt of the acknowledgement of the first packet in the acknowledgement queue, when the number of unacknowledged packets is equal to the window size. In the analytical model, the Markovian assumption on the acknowledgement intervals is made. This means the blocking experienced by a blocked packet is on the average, mean acknowledgement interval. This is because the residual service time of the first packet at stage 2 is independent of the time already spent in service.

CHAPTER 5

DISCUSSION OF RESULTS

5.1 ACCURACY OF THE MODEL

The results obtained by simulation and analysis (Alg. 3.2) are listed in Table 5.1. Results of a second method of analysis, which incorporates an approximation to account for the hyperexponential nature of the server at stage 1 are also listed. Fig. 5.1 clearly indicates the need for the second method under moderate traffic. This is further supported by results of Appendix B (eqn. B.6). The approximation is explained in Sec. 5.2.

The analytical methods are able to predict reasonably well the parameters of interest at a computer network node, namely, throughput, delay, network power. As explained the second method gives better results. Also the buffer overflow probabilities are approximated reasonably well by the second method. The first method is not so accurate for this parameter. It may be seen from Figures 5.1 - 5.5 that in most cases the two analytical methods outline the envelope in which the actual values, given by simulation, lie.

The mean and standard deviation of the effective service times (\bar{x} , σ_x) and interdeparture times (\bar{d} , σ_d) at stage 1 are predicted fairly accurately, in most cases. \bar{x} ($=1/\mu_{\text{eff}}$)

and \bar{d} ($= 1/\gamma$) are given by algorithm 3.2. σ_x is found using the coefficient of variation, C_b , of the service time, which is given by eqn. C.18 (Appendix C). We have omitted the effect of p_2 (eqn. C.18). This becomes appreciable only for W in the range 2-3.

The coefficient of variation of the interdeparture times, C_d , is used to determine σ_d . C_d is approximated by the result for the interdeparture times in an $M/G/1/\infty$ system [MARS 68],

$$C_d = 1 + \rho^2(C_b^2 - 1) \quad (5.1.1)$$

where

$$\rho = \lambda \bar{X}$$

5.2 APPROXIMATIONS TO ACCOUNT FOR HYPEREXPONENTIAL SERVER

The stage 1 server, effectively being hyperexponential (Appendix C, eqn. C.16) makes the 2 stage model $M/H_2/1/N \rightarrow . /M/1/W$, with successful transmission of packets, corresponding to departures from stage 1. Two approximations are used :

- i) The second stage is effectively an $H_2/M/1/W$ system, due to output of stage 1 being hyperexponential (eqn. 5.1.1). The coefficient of variation of the arrival process to second stage is used to calculate the parameters of the H_2 distribution (eqns. A.1, A.5).

The traffic intensity, σ , is found by eqn. B.4. The $M/M/1/W$ system analysis is used to calculate the queue length with this traffic intensity, σ . The approximation made is the increase in queue length due to greater variation in service time is incorporated as increase in traffic intensity in an $M/M/1/W$ system.

- ii) The hyper exponential server at stage 1 gives rise to waiting times greater than that for an exponential server with the same mean. The first stage is effectively $M/H_2/1/N$. However, $M/G/1$ analysis is more difficult than $GI/M/1$ analysis as pointed out in Sec. 2.3. We incorporate the effect of increase in queue length due to the increased variation in service time as if the $M/M/1/N$ system was having a higher traffic intensity. We use the $H_2/M/1/\infty$ correction again (eqn. D.16) again. To do this, we reflect the coefficient of variation of the service time at stage 1 to the arrival process and assume the service time is exponentially distributed.

Considerable approximations have been made. These require careful scrutiny. A search in literature, or analysis, for queue lengths for $M/H_2/1/K$ and $H_2/M/1/K$ systems, and coefficient of variation of interdeparture times from an $M/H_2/1/K$ may be done for better results.

5.3 SENSITIVITY OF POWER TO VARIATIONS IN PREDICTED VALUES OF PARAMETERS

The network power, referring to notations of Chapter 3, is,

$$P = \frac{\text{Throughput}}{\text{delay}}$$

or

$$P = \frac{\gamma}{Q_1 / \gamma} = \frac{\gamma^2}{Q_1} \quad (5.2.1)$$

P is therefore more sensitive to errors in the prediction of γ . It may be seen that $\gamma = 1/\bar{d}$, is predicted reasonably accurately in most cases. The error in prediction of P, in most cases, is due to the incorrect prediction of L.

- i) This may be due to the assumption of buffer occupancy by unacknowledged packets small enough to be neglected (Sec. 3.1). The error introduced by this is not expected to be substantial, simulation results show the expected occupancy by unacknowledged packets Q_2 , do not exceed 3.

With buffer sizes of the order of 30 or more the error in N for queue length calculation (eqn. 2.2.4) is not more than 10 percent. Furthermore the analytical methods underestimate Q_2 . This appears to be due to the approximations used.

- ii) The main cause for discrepancies in calculation of Q_1 is due to the approximations made to incorporate the effect of H_2 server at stage 1.

It may be noted that the correction factor for the hyperexponential server is not incorporated in the second method for traffic intensities at stage 1, $\rho_1 > 1$. This is because the result derived is for $M_2/M/1/\infty$ system, where the ergodicity condition is $\rho_1 < 1$. Studies may be done to see whether the correction factor is significant for $\rho_1 > 1$. We do not expect this to be significant due to heavy traffic. At $\rho_1 = 1$, the correction factor is zero (Appendix B).

5.4 DESIGN CHOICE OF BUFFER SIZE AND WINDOW

The second method is found to be more accurate in predicting the buffer overflow probability, p_1 . This may be used to find a proper buffer size, N .

A plot of $p_1 \gamma_s N$ is shown in figure 5.2. Plot of power (P) vs N would give false notion of increase in power with reduction in N . This is due to reduced queue lengths and hence delays. A node however cannot be considered in isolation. Smaller N values give rise to higher p_1 values. This means increase in retransmissions and hence delay, at previous node.

The choice of N should be one that reduces buffer overflow probability within acceptable limits. At the same time, N should not be too large. The response to congestion would be sluggish. It may be noted (Chapter 1) that information (effect) of congestion in an interior node must be passed to the

entry points as quickly as possible for good network flow control scheme.

Variation of power vs W is shown in Fig. 5.3. There is no optimal W . However, beyond a certain range, it is seen that the increase in power with W is very small. W must be chosen just sufficiently large to give the desired performance. A substantially large W increases the probability of congestion.

Furthermore the model allows a calculation of the upper bound of the blocking probability, p_B due to window. This happens when the traffic intensity (note traffic is already regulated by stage 1 and normally stage 2 server is faster than stage 1 server) at stage 2, ρ_2 is 1. p_B is then $\frac{1}{W+1}$. For $W = 8$ (.e.g. HDLC), $p_B = 1/9$. This is strictly an upper bound. ρ_2 is actually much less than 1.

It is further observed that the values of N and W on the performance are correlated. A small window would restrict the throughput irrespective of the buffer size. It is observed from the results that beyond a certain limit e.g. $W = 5$ increase of window increases power as 'a fine tuning mechanism'. The choice of $W = 8$ for HDLC is thus a good one.

The design procedure for N and W is thus :

Given traffic parameters, time out, retransmission probabilities.

- i) Initialize W between 5 to 8.
- ii) Increase N till approximately the acceptable buffer over-flow probability is attained.
- iii) Increase W for better performance, at the same time keep within reasonable limits for congestion avoidance.

5.5 OPTIMAL TIME OUT FACTOR

The analytical model predicts the performance with various timeout intervals close to simulation. Some cases are plotted in Fig. 5.4. A small time out will give rise to greater probability of retransmission. A large time out will degrade the performance of the transmission stage.

Referring to sec, 3.5.,

$$\mu_{1 \text{ eff}} = \frac{\mu_1}{1 + (i+1) \frac{p}{1-p}}$$

where

$$p = p_e + p_r$$

p_r given by Appendix D (eqn. D.6), is a strictly decreasing function of i . The optimal value of i such that $\mu_{1 \text{ eff}}$ is maximized may be found analytically or graphically.

It was observed that for $W > 10$, approximation for $W = \infty$ (Appendix D) may be used.

5.6 INDEPENDENCE OF OUTPUTS

It was observed that the autocorrelation coefficients ($\text{lag} > 1$) for departures from the transmission stage (stage 1) were sufficiently small. This allows assumption of independence of the outputs from engineering viewpoints. This is useful for extension of analysis to network of queues. The inputs to the next node may thus be assumed independent.

This is an important observation because of Finch's result [FINC 59] that the output of an M/G/1/K system is independent only if the server is exponential and the waiting room, $K = \infty$. We may look into the method of proof for correlation of interdeparture times in an M/M/1/N system to get an insight to the observed results.

Finch showed that, for the output to be independent, the probability of a departure leaving no customers behind,

$$p_0 = 1 - \lambda \bar{X} \quad (5.6.1)$$

λ , \bar{X} being the notations introduced earlier. However, M/M/1/N system analysis leads to

$$p_0 = (1 - \rho) / (1 - \rho^{N+1}) \quad (5.6.2)$$

where $\rho = \lambda \bar{X}$.

This violates eqn. (5.6.1) and hence the original assumption of independence was false. However, from engineering considerations, for large N (> 20) the denominator in eqn. (5.6.2) is sufficiently close to unity, particularly if ρ is not close to unity.

Furthermore, the deviation of coefficient of variation of the server at stage 1 is not substantially different from unity, as shown by the results.

Also the coefficient of variation of the interdeparture times is bounded by the coefficient of variation of the service time, if the arrivals are poisson (eqn. 5.1.1). The upper bound occurs only when the server is busy at all times ($\lambda = \infty$). Thus independent arrivals with poisson distribution may be assumed for the next node.

Further work on the autocorrelation coefficients may be done based e.g. on [DALE 68] or Cox's result [BURK 72]

$$\sum_{r=1}^{\infty} a_r = \frac{\rho^2(1-C^2)}{2[1-\rho^2(1-C^2)]} \quad (5.6.3)$$

where a_r = autocorrelation coefficient of lag r

C = coefficient of variation of service time

$$\rho = \lambda \bar{X}$$

5.7 REPLACEMENT OF EFFECTIVE TRANSMISSION PROCESS BY A HYPEREXPONENTIAL SERVER

The transmission stage (server 1) with retransmissions after time out is modelled by an equivalent hyperexponential server (eqns. C.16, A.4).

Simulations were carried out, replacing the transmission stage including retransmissions after time out mechanism, by the equivalent hyperexponential server. The mean and variance of service time of the server was fed from the results of simulations with the retransmission mechanism. It was observed that the results were accurate to those of the original model. The blocking probability by server 2 is slightly higher with the replacement by hyperexponential server.

This has two consequences :

- i) Any complex transmission/retransmission mechanism may be given an equivalent H_2 representation. Queuing systems analysis require only the mean and variance. Approximating to the n moments requires an H_n representation. Hence, the H_2 representation is sufficient.
- ii) Considerable saving in computer time results by replacing the complex retransmission procedures in simulations, particularly when simulating network of queues.

This however requires a good approximation for the mean and variance of each transmission stage in the network. This may be taken up for further study.

Simulations were also performed under Erlang-3 assumptions on packet transmission times. This is based on physical packet length having less variation than that given by the exponential distribution. This stage, with coefficient of variation < 1 , in series with the retransmission stage, coefficient of variation > 1 , gives overall performance closer to exponential. The $M/M/1/N \rightarrow M/M/1/W$ analysis was found more accurate in this case.

Simulations were also performed varying λ , μ_1 and μ_2 but keeping the ratios fixed. The parameters which lead to calculations of delay, throughput etc., namely, p_1, p_2, Q_1, Q_2, p_r were quite close. This was performed to check for the effect of $\mu_1 \text{ eff}$ (sec 3.5). This gives rise to $\rho_1 = \lambda / \mu_1 \text{ eff}$, not strictly dependent on the the ratios. The explanation is that for most cases p_2 is sufficiently small. It is expected that for p_2 large (e.g. $W = 2-3$) change of the parameters λ_1, μ_1, μ_2 with ratios constant may give results that are not very close.

Under operating conditions of $W > 5$, the results in Table 5.1 may therefore, be used to infer performance measures for other ranges of values of λ, μ_1, μ_2 .

5.8 STABILITY OF THE NUMERICAL METHOD

It was observed that, for the cases considered, convergence was independent of the starting values of p_2 and p_r . It was found that in some cases, however, convergence was not possible because of oscillations in the values of p_r . It was observed, in most of these cases that the mean of the two values between which p_r was oscillating, was close to the actual value, given by simulation. These cases are listed below :

- i) $\lambda = 7.0$, $\mu_1 = 10.0$, $\mu_2 = 10.0$, $N = 15$, $W = 10$, $i = 5$
for values of $p_e < 0.05$.
- ii) $\lambda = 40.0$, $\mu_1 = 50.0$, $\mu_2 = 50.0$, $p_e = 0.01$, $N = 15$, $W = 10$,
 $i = 5$.

5.9 SOME OTHER POSSIBLE AREAS FOR FURTHER WORK

An area which needs attention is the modelling of the acknowledge process (stage 2). With piggy backed ACK schemes, and acknowledging upto K previous packets e.g. HDLC, it is felt that exponential server with bulk service [KLEI 75] would be a better model.

Analysis for state dependent processes can be done by incorporation in the current numerical procedure. These can be useful example determining loss probability, of voice packets where the probability of arrival of a packet and duration of service may be

state dependent on the number of packets already present (no. of active talkers).

Extension of dynamic flow control measures based on Markovian Decision model of Kemani and Kleinrock [KERM 80a] may perhaps be done upto link level. It may be interesting to see whether a dynamic control on link level window, time outs would actually improve the performance as in end-to-end case.

The effective transmission time may be interpreted due to increased average packet lengths and no retransmission. A comparison with error coding methods e.g. forward-error-correction for random or burst errors, may be done for noisy channels.

Finally, the extension to the overall performance of a network needs to be done. Direct application of Jackson's theorem, BCMP queues are not applicable due to finite queuing assumptions. Without this assumption the effects of retransmissions are lost.

A generalized numerical procedure extending the present one for a single link may be designed.

For example for a tandem network we would have to modify algorithm 3.2 to an algorithm that would be an extension of the method of Irland et al [IRLA 77].

Now there are two levels of convergence. The outer loop would account for the nodal blocking probabilities for the tandem physical nodes. For n nodes start with $p_{1j} = 0, j = 1, \dots, n$. p_{1j} corresponds to p_1 determined by algorithm 3.2, for node j .

If the arrival rate to the network λ , then the effective arrival rates to individual nodes get modified due to retransmission to [IRLA 77]

$$\lambda_1 = \lambda, \quad \lambda_j = \frac{\lambda_{1j-1}}{(1-p_{1j})}, \quad j = 2, \dots, n$$

Using this value of λ_j for λ in Algorithm 3.2, we now iterate for stability for each node. Note that at the beginning of the iteration of the inner loop (Algorithm 3.2) for individual node,

$$p_{ej} = p_{Lj} + p_{1j}$$

where p_{Lj} is the physical packet or acknowledgement error rate, on link out of node j .

Thus, we propose the following numerical procedure for tandem network.

Algorithm 5.3

Given a network of n nodes in tandem with arrival rate

λ , time out factors i_j ,

line loss rates p_{Lj} , transmission rates μ_{1j} , acknowledgement

rates μ_{2j} , buffer sizes N_j , window sizes W_j , $j = 1, 2, \dots, n$.

Initialize $p_{1j} = 0$, $j = 1, \dots, n$,

Repeat

For $j=1, 2, \dots, n$ do begin

If $j=1$ $\lambda_j = \lambda$ else $\lambda_j = \lambda_{1,j-1}/(1-p_{1j})$

(* Algorithm 3.2 *)

Initialize $p_{rj} = p_{2j} = 0$, $j = 1, \dots, n$.

$p_{ej} = p_{1j} + p_{1j}$.

While not stable p_{rj} and p_{2j} do
begin

$p_j = p_{ej} + p_{rj}$

$$\mu_{1loss,j} = \frac{\mu_{1j}}{1 + (i_j + 1) \frac{p_j}{1 - p_j}}$$

$$\mu_{1eff,j} = \frac{\mu_{1loss,j} \mu_{2j}}{\mu_{2j} + p_{2j} \mu_{1loss,j}}$$

$$\rho_{1j} = \rho_j / \mu_{1eff,j}$$

$$\rho_{oj} = (1 - \rho_{1j}) / (1 - \rho_{1j}^{N_j + 1})$$

$$\lambda_{1j} = \mu_{1eff,j} (1 - \rho_{oj})$$

$$\lambda_{2j} = \lambda_{1j} / (1 - p_{2j}) \quad ; \quad \rho_{2j} = \lambda_{2j} / \mu_{2j}$$

$$p_{2j} = (1 - p_{2j})^{p_{2j}^{w_j}} / (1 - p_{2j}^{w_j+1})$$

Calculate p_{rj} by eqn. D.6 (Appendix D).

end

$$p_{1j} = p_{0j} p_1^N$$

end; until p_{1j} 's stabilize

for $j = 1, \dots, n$

begin

 calculate L_j by eqn.(2.2.4) using p_{1j}, N_j

$$D_j = L_j / \lambda'_j \quad \text{where } \lambda'_j = \lambda_{1j} = \text{system throughput}$$

end.

5.10 SUMMARY

Performance evaluation of existing flow control schemes or choice of parameters for optimal performance for a new scheme are required in computer networks. Simulation offers one solution but requires large amount of computer time. Exact methods, based on queuing theory are complex for computation purposes. Furthermore, the effort spent in exact solutions may be futile from engineering viewpoints due to the discrete values of physical components e.g. transmission line capacities, timeout intervals etc. The need arises, therefore, for simpler approximate methods.

The approximation method (based on a 2-stage queuing model), presented in this thesis, is able to predict the performance parameters reasonable well. Furthermore, each control parameter is given an interpretation in terms of queuing effects e.g. retransmission after finite time out increases the coefficient of variation of the transmission stage, the acknowledgement process reduces the coefficient of variation of the transmission stage, window limits the waiting room in front of the second stage etc.

legend for table 5.1 and figs. 5.1-5.5

- (a) - results of simulation
- (b) - results of analysis (alg. 3.2)
- (c) - results of analysis with correction for H_2 server at stage 1

\bar{x}, σ_x - mean, st. dev. of effective service time at stage 1

\bar{d}, σ_d - mean, st. dev of interdep. times from node.

\bar{D}, σ_D - mean, st. dev. of delay at physical node

ρ - network power ($= \lambda / \bar{D} = 1 / (\bar{d} \cdot \bar{D})$)

ρ_1 - nodal buffer overflow probability

ρ_2 - prob. of blocking of packet due to window

ρ_e - probability of retransmission due to transmission errors, buffer overflow condition at next node

N - physical buffer size

M - max. no. of unacknowledged packets permissible

ρ_1 - average nodal buffer occupancy

ρ_2 - average buffer occupancy by unack'd packets

ρ_r - probability of redundant retransmission

———— plot of simulation results

----- plot of analytical results (b)

----- plot of analytical results (c)

(results of (b) not plotted if close to (c) ;
analytical results not shown if close to siml.)

TABLE 5.1
RESULTS OF SIMULATION AND ANALYSIS

(a) variation with λ

parameters	\bar{x}	σ_x	\bar{a}	σ_d	\bar{D}	σ_D	P	p1	p2	q1	q2	p _r
$\lambda = 5, 0, \mu_1 = 90, 0, \mu_2 = 90, 0, \text{Pe} = 0.1, N = 25, w = 5, l = 5$	(a) (b) (c)	0.18 0.18 0.13	0.027 0.026 0.026	0.197 0.2 0.2	0.199 0.2 0.149	0.025 0.02 0.027	229.68 244.44 183.02	0.0 0.0 0.0	0.0 0.0 0.0	0.165 0.102 0.136	0.154 0.058 0.077	0.002 0.001 0.001
$\lambda = 10, 0, \mu_1 = 90, 0, \mu_2 = 90, 0, \text{Pe} = 0.1, N = 25, w = 5, l = 5$	(a) (b) (c)	0.319 0.319 0.313	0.027 0.026 0.026	0.1 0.1 0.1	0.102 0.102 0.09	0.03 0.022 0.031	385.0 436.0 321.22	0.0 0.0 0.0	0.0 0.0 0.0	0.373 0.228 0.331	0.319 0.128 0.168	0.012 0.001 0.002
$\lambda = 25, 0, \mu_1 = 90, 0, \mu_2 = 90, 0, \text{Pe} = 0.1, N = 25, w = 5, l = 5$	(a) (b) (c)	0.019 0.02 0.02	0.027 0.027 0.027	0.04 0.04 0.04	0.044 0.044 0.039	0.061 0.036 0.055	450.88 681.11 450.88	0.0 0.0 0.0	0.003 0.001 0.004	1.617 0.917 1.386	0.92 0.382 0.55	0.007 0.007 0.013
$\lambda = 35, 0, \mu_1 = 90, 0, \mu_2 = 90, 0, \text{Pe} = 0.1, N = 25, w = 5, l = 5$	(a) (b) (c)	0.02 0.02 0.02	0.028 0.028 0.031	0.028 0.029 0.029	0.034 0.035 0.034	0.103 0.060 0.12	372.05 526.65 289.58	0.001 0.0 0.001	0.013 0.006 0.015	3.926 2.325 4.221	1.418 0.62 0.886	0.021 0.016 0.03
$\lambda = 40, 0, \mu_1 = 90, 0, \mu_2 = 90, 0, \text{Pe} = 0.1, N = 25, w = 5, l = 5$	(a) (b) (c)	0.021 0.022 0.022	0.029 0.03 0.031	0.025 0.025 0.025	0.031 0.032 0.033	0.152 0.114 0.21	242.07 349.18 187.52	0.009 0.002 0.012	0.025 0.01 0.022	7.286 4.568 8.328	1.69 0.73 1.04	0.036 0.023 0.0378
$\lambda = 55, 0, \mu_1 = 90, 0, \mu_2 = 90, 0, \text{Pe} = 0.1, N = 25, w = 5, l = 5$	(a) (b) (c)	0.021 0.022 0.022	0.028 0.031 0.031	0.021 0.022 0.022	0.028 0.031 0.031	0.161 0.439 0.439	120.23 104.41 104.41	0.151 0.166 0.166	0.039 0.019 0.019	18.966 20.159 20.159	2.029 0.96 0.96	0.04 0.033 0.033

table 5.1 contd./

(a) variation with w

parameters	\bar{x}	σ_x	\bar{d}	σ_d	\bar{D}	σ_D	r	p_1	p_2	Q_1	Q_2	p_r
$\lambda = 40, 0, \mu_1 = 95, 0, \mu_2 = 100, 0, \rho = 0.1, N = 10, w = 5, l = 5$	(a) (b) (c)	0.019 0.019 0.022	0.026 0.026 0.028	0.026 0.025 0.026	0.031 0.032 0.032	0.075 0.063 0.088	507.06 615.97 434.56	0.04 0.015 0.035	0.012 0.006 0.014	3.424 2.522 3.43	1.387 0.633 0.87	0.023 0.014 0.024
$\lambda = 40, 0, \mu_1 = 95, 0, \mu_2 = 100, 0, \rho = 0.1, N = 15, w = 5, l = 5$	(a) (b) (c)	0.019 0.019 0.02	0.026 0.026 0.028	0.025 0.025 0.025	0.031 0.031 0.031	0.096 0.071 0.111	405.84 554.15 352.22	0.02 0.004 0.011	0.015 0.006 0.016	4.39 2.87 4.41	1.46 0.644 0.897	0.026 0.015 0.025
$\lambda = 40, 0, \mu_1 = 95, 0, \mu_2 = 100, 0, \rho = 0.1, N = 20, w = 5, l = 5$	(a) (b) (c)	0.019 0.019 0.02	0.026 0.026 0.028	0.025 0.025 0.025	0.03 0.031 0.031	0.104 0.074 0.126	380.15 534.24 313.58	0.006 0.001 0.006	0.016 0.01 0.016	4.753 2.985 5.037	1.489 0.647 0.908	0.026 0.015 0.026
$\lambda = 40, 0, \mu_1 = 95, 0, \mu_2 = 100, 0, \rho = 0.1, N = 25, w = 5, l = 5$	(a) (b) (c)	0.019 0.019 0.02	0.026 0.026 0.028	0.025 0.025 0.025	0.031 0.031 0.031	0.112 0.075 0.135	354.59 527.97 293.9	0.003 0.0 0.003	0.016 0.001 0.016	5.082 3.029 5.412	1.493 0.04 0.913	0.026 0.015 0.026

table 5.1 contd/.

(c) variation with *

parameters	\bar{x}	σ_x	\bar{u}	σ_u	\bar{D}	σ_D	P	p1	p2	Q1	Q2	p _r
$\lambda=80, \mu=95, \sigma,$ $\mu_2=100, \sigma, \rho=0.1,$ $N=80, w=2, i=10$	(a) (b) (c)	0.024 0.028 0.026	0.044 0.047 0.047	0.029 0.026 0.026	0.045 0.042 0.047	2.241 2.064 2.064	0.469	0.559 0.521 0.521	0.437 0.116 0.116	78.324 79.081 79.081	1.437 0.498 0.498	0.0 0.0 0.0
$\lambda=80, \mu=95, \sigma,$ $\mu_2=100, \sigma, \rho=0.1,$ $N=80, w=5, i=10$	(a) (b) (c)	0.023 0.024 0.024	0.041 0.043 0.043	0.024 0.024 0.024	0.041 0.043 0.043	1.858 1.864 1.864	0.382	0.477 0.47 0.47	0.008 0.008 0.01	78.164 78.878 78.878	2.353 0.707 0.707	0.002 0.0 0.0
$\lambda=80, \mu=95, \sigma,$ $\mu_2=100, \sigma, \rho=0.1,$ $N=80, w=10, i=10$	(a) (b) (c)	0.023 0.024 0.024	0.043 0.042 0.042	0.023 0.024 0.024	0.043 0.042 0.042	1.833 1.855 1.855	0.407	0.471 0.469 0.469	0.004 0.0 0.0	78.122 78.866 78.866	2.704 0.738 0.738	0.005 0.001 0.001
$\lambda=80, \mu=95, \sigma,$ $\mu_2=100, \sigma, \rho=0.1,$ $N=80, w=15, i=10$	(a) (b) (c)	0.024 0.024 0.024	0.043 0.043 0.043	0.024 0.024 0.024	0.043 0.043 0.043	1.865 1.856 1.856	0.407	0.478 0.469 0.469	0.0 0.0 0.0	78.103 78.878 78.867	2.63 0.738 0.738	0.005 0.001 0.001

Table S.1 (c) contd/

parameters	\bar{x}	δ_x	\bar{a}	δ_a	\bar{b}	σ_b	p	p1	p2	Q1	Q2	p	r
$\lambda = 40, \mu = 95, \sigma = 0.03, N = 30, w = 2, l = 10$	(A) 0.026	0.037	0.026	0.039	0.379	0.261	100.2	0.048	0.42	14.914	1.382	0.0	
	(B) 0.025	0.045	0.025	0.043	0.275		168.19	0.008	0.124	9.359	0.52	0.0	
	(C) 0.026	0.045	0.026	0.046	0.376		103.17	0.028	0.202	14.644	0.716	0.0	
$\lambda = 40, \mu = 95, \sigma = 0.03, N = 30, w = 5, l = 10$	(A) 0.022	0.039	0.025	0.039	0.198	0.194	199.13	0.011	0.06	8.64	1.963	0.002	
	(B) 0.022	0.038	0.025	0.041	0.147		271.63	0.002	0.044	5.87	0.591	0.0	
	(C) 0.021	0.038	0.025	0.04	0.234		169.14	0.008	0.029	9.31	1.153	0.001	
$\lambda = 40, \mu = 95, \sigma = 0.03, N = 30, w = 10, l = 10$	(A) 0.021	0.037	0.025	0.039	0.197	0.2	199.0	0.01	0.001	8.933	2.16	0.003	
	(B) 0.021	0.038	0.025	0.04	0.117		338.98	0.0	0.0	4.714	0.665	0.006	
	(C) 0.021	0.038	0.025	0.04	0.232		170.82	0.008	0.002	9.221	1.305	0.005	
$\lambda = 40, \mu = 95, \sigma = 0.03, N = 30, w = 15, l = 10$	(A) 0.02	0.037	0.025	0.04	0.196	0.195	197.49	0.012	0.0	8.602	2.086	0.003	
	(B) 0.021	0.037	0.025	0.04	0.117		338.81	0.001	0.0	4.716	0.664	0.001	
	(C) 0.021	0.039	0.025	0.04	0.237		167.37	0.008	0.0	9.402	1.317	0.006	

table 5.1(c) contd./

parameters	\bar{x}	σ_x	\bar{u}	σ_u	\bar{D}	σ_D	P	p1	p2	Q1	Q2	P _T
$\lambda = 40, 0, \mu_1 = 95, 0, \mu_2 = 100, 0, \mu_3 = 20, w = 2, 1 = 5$ (A) (B) (C)	0.016 0.018 0.002	0.023 0.025 0.027	0.025 0.025 0.025	0.028 0.03 0.03	0.103 0.07 0.114	0.104	379.89 570.05 348.07	0.006 0.001 0.005	0.315 0.125 0.175	4.56 2.803 4.555	1.165 0.525 0.653	0.012 0.005 0.007
$\lambda = 40, 0, \mu_1 = 95, 0, \mu_2 = 100, 0, \mu_3 = 20, w = 5, 1 = 5$ (A) (B) (C)	0.017 0.017 0.016	0.024 0.024 0.026	0.025 0.025 0.025	0.029 0.03 0.029	0.075 0.055 0.095	0.081	542.92 723.44 420.04	0.002 0.002 0.002	0.012 0.0 0.016	3.496 2.210 3.791	1.372 0.647 0.913	0.023 0.015 0.026
$\lambda = 40, 0, \mu_1 = 95, 0, \mu_2 = 100, 0, \mu_3 = 20, w = 1, 1 = 5$ (A) (B) (C)	0.017 0.017 0.019	0.024 0.024 0.036	0.025 0.025 0.025	0.029 0.03 0.03	0.075 0.055 0.099	0.083	523.52 716.42 399.12	0.002 0.0 0.003	0.0 0.001 0.004	3.561 2.232 3.986	1.39 0.669 0.961	0.025 0.017 0.032
$\lambda = 40, 0, \mu_1 = 95, 0, \mu_2 = 100, 0, \mu_3 = 20, w = 10, 1 = 5$ (A) (B) (C)	0.017 0.017 0.019	0.023 0.024 0.026	0.025 0.025 0.025	0.029 0.03 0.03	0.076 0.056 0.102	0.082	523.33 713.62 389.32	0.002 0.0 0.003	0.0 0.0 0.0	3.558 2.241 4.08	1.406 0.666 0.981	0.025 0.017 0.034

table 5.1 contd/

(d) variation with λ

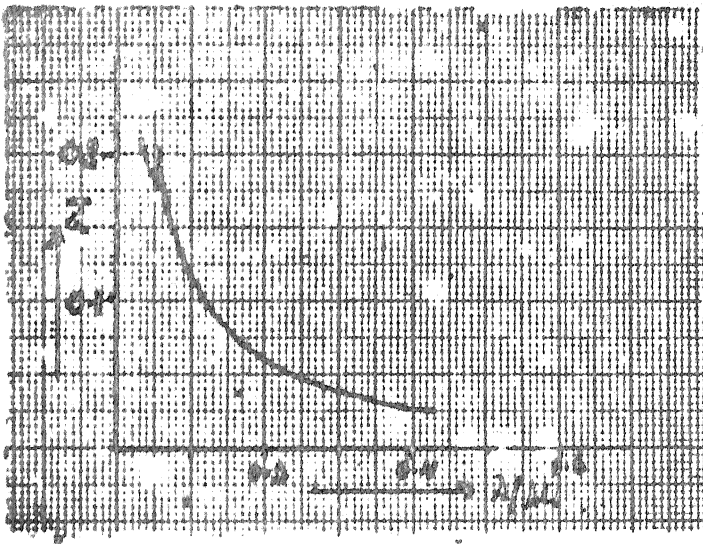
parameters	\bar{x}	σ_x	\bar{a}	σ_a	\bar{b}	σ_b	P	p1	p2	Q1	Q2	p_r
$\lambda = 30.0, \mu = 50.0,$ $\mu_s = 70.0, \rho = 0.08,$ $N = 15, n = 5, l = 2$	(a) 0.031 (b) 0.031 (c) 0.032	0.034 0.034 0.035	0.035 0.035 0.035	0.036 0.038 0.038	0.221 0.215 0.234	0.161	127.8 134.12 121.71	0.048 0.038 0.047	0.001 0.007 0.009	6.7 6.211 6.7	1.311 0.679 0.728	0.123 0.076 0.084
$\lambda = 30.0, \mu = 50.0,$ $\mu_s = 70.0, \rho = 0.08,$ $N = 15, n = 5, l = 3$	(a) 0.03 (b) 0.03 (c) 0.031	0.035 0.036 0.037	0.034 0.034 0.035	0.039 0.04 0.04	0.196 0.191 0.221	0.153	146.86 152.18 129.74	0.041 0.028 0.041	0.006 0.008 0.011	6.249 5.583 6.383	1.462 0.69 0.794	0.045 0.032 0.04
$\lambda = 30.0, \mu = 50.0,$ $\mu_s = 70.0, \rho = 0.08,$ $N = 15, n = 5, l = 5$	(a) 0.031 (b) 0.031 (c) 0.032	0.044 0.043 0.044	0.035 0.035 0.035	0.047 0.045 0.047	0.224 0.215 0.243	0.181	124.13 133.85 116.92	0.023 0.038 0.051	0.023 0.007 0.015	6.908 6.222 6.935	1.678 0.679 0.891	0.008 0.005 0.082
$\lambda = 30.0, \mu = 50.0,$ $\mu_s = 70.0, \rho = 0.08,$ $N = 15, n = 5, l = 7$	(a) 0.034 (b) 0.034 (c) 0.034	0.054 0.053 0.053	0.038 0.036 0.036	0.056 0.053 0.056	0.289 0.29 0.29	0.226	90.57 95.38 95.38	0.124 0.076 0.076	0.037 0.006 0.006	8.335 8.061 8.061	1.792 0.638 0.638	0.002 0.006 0.006

table 5.1 (a) contd/

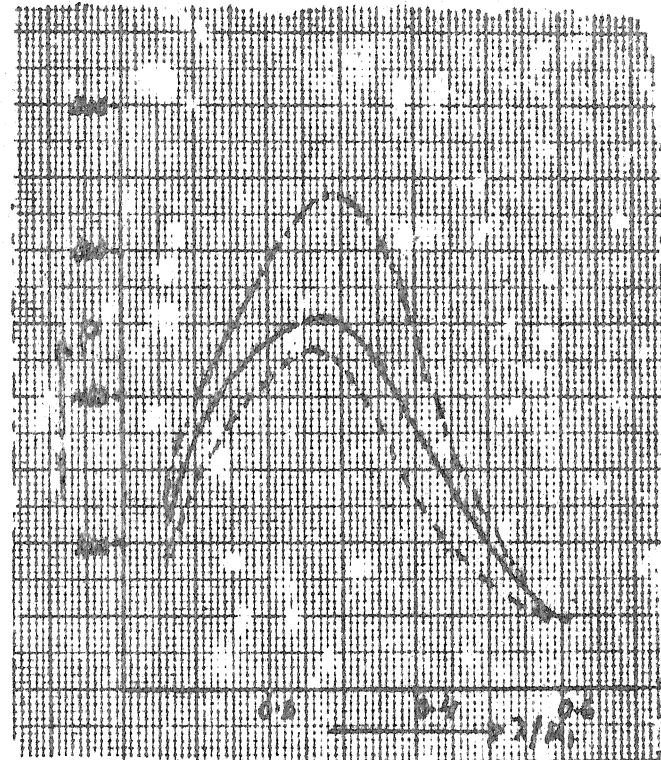
parameters	\bar{x}	σ_x	\bar{y}	σ_y	\bar{D}	σ_D	P	p1	p2	Q1	Q2	P _r
$\lambda = 6.0, \mu_1 = 7.0, \mu_2 = 10.0, \rho = 0.0, N = 15, n = 5, l = 1$	(A) (D) (E)	0.255 0.22 0.22	0.236 0.22 0.22	0.255 0.22 0.22	3.248 2.643 2.643	1.110	1.2 1.72 1.72	0.343 0.242 0.242	0.0 0.011 0.011	13.074 12.021 12.021	1.227 0.794 0.794	0.45 0.205 0.205
$\lambda = 5.0, \mu_1 = 7.0, \mu_2 = 10.0, \rho = 0.0, N = 15, n = 5, l = 5$	(A) (D) (E)	0.148 0.158 0.151	0.16 0.195 0.18	0.172 0.174 0.175	0.898 1.12 1.198	0.679	6.44 5.13 4.76	0.034 0.042 0.049	0.035 0.03 0.037	6.295 6.44 6.84	1.86 1.185 1.27	0.009 0.013 0.014
$\lambda = 6.0, \mu_1 = 7.0, \mu_2 = 10.0, \rho = 0.0, N = 15, n = 5, l = 5$	(A) (D) (E)	0.142 0.148 0.148	0.142 0.148 0.148	0.172 0.17 0.17	0.8 0.869 0.871	0.638	7.23 6.75 6.73	0.025 0.022 0.022	0.053 0.034 0.034	5.869 5.1 5.1	1.947 1.23 1.23	0.0 0.0 0.0
$\lambda = 6.0, \mu_1 = 7.0, \mu_2 = 10.0, \rho = 0.0, N = 15, n = 5, l = 15$	(A) (D) (E)	0.142 0.148 0.148	0.141 0.148 0.148	0.172 0.17 0.17	0.798 0.866 0.866	0.637	7.25 6.77 6.77	0.024 0.022 0.022	0.051 0.034 0.034	5.814 5.086 5.086	1.922 1.23 1.23	0.0 0.0 0.0

table 5.1 (a) contd./

parameters	\bar{x}	σ_x	\bar{z}	σ_z	\bar{y}	σ_y	P	p1	p2	o1	Q2	p _r
$\lambda=30, \mu=50, \sigma=0.1, \nu=15, w=5, l=5$	(a) 0.0259 (b) 0.0255 (c) 0.025	0.030 0.031 0.031	0.034 0.034 0.035	0.037 0.042 0.046	0.125 0.154 0.231	0.12	233.58 190.6 123.48	0.015 0.016 0.045	0.027 0.035 0.053	4.669 4.571 6.643	1.727 1.244 1.483	0.04 0.051 0.066
$\lambda=30, \mu=50, \sigma=0.1, \nu=15, w=5, l=7$	(a) 0.023 (b) 0.025 (c) 0.027	0.03 0.03+ 0.036	0.033 0.033 0.034	0.037 0.04 0.042	0.097 0.096 0.162	0.103	304.09 311.25 181.69	0.011 0.004 0.018	0.047 0.037 0.067	4.078 2.871 4.774	1.86 1.273 1.641	0.011 0.016 0.024
$\lambda=30, \mu=50, \sigma=0.1, \nu=15, w=5, l=10$	(a) 0.022 (b) 0.024 (c) 0.025	0.031 0.032 0.035	0.033 0.033 0.034	0.039 0.04 0.04	0.098 0.078 0.122	0.11	299.88 383.14 241.96	0.014 0.002 0.008	0.073 0.037 0.073	4.204 2.341 3.659	1.969 1.277 1.709	0.002 0.002 0.004
$\lambda=30, \mu=50, \sigma=0.1, \nu=15, w=5, l=12$	(a) 0.022 (b) 0.024 (c) 0.025	0.034 0.034 0.036	0.033 0.033 0.037	0.041 0.042 0.041	0.105 0.078 0.127	0.118	279.28 379.65 233.42	0.019 0.002 0.009	0.076 0.037 0.079	4.455 2.363 3.786	2.014 1.277 1.772	0.0 0.0 0.0
$\lambda=30, \mu=50, \sigma=0.1, \nu=15, w=5, l=15$	(a) 0.023 (b) 0.024 (c) 0.025	0.037 0.04 0.04	0.034 0.033 0.034	0.044 0.045 0.045	0.115 0.084 0.148	0.133	250.09 354.77 199.08	0.027 0.002 0.014	0.082 0.037 0.091	4.697 2.525 4.392	2.038 1.276 1.888	0.0 0.0 0.0



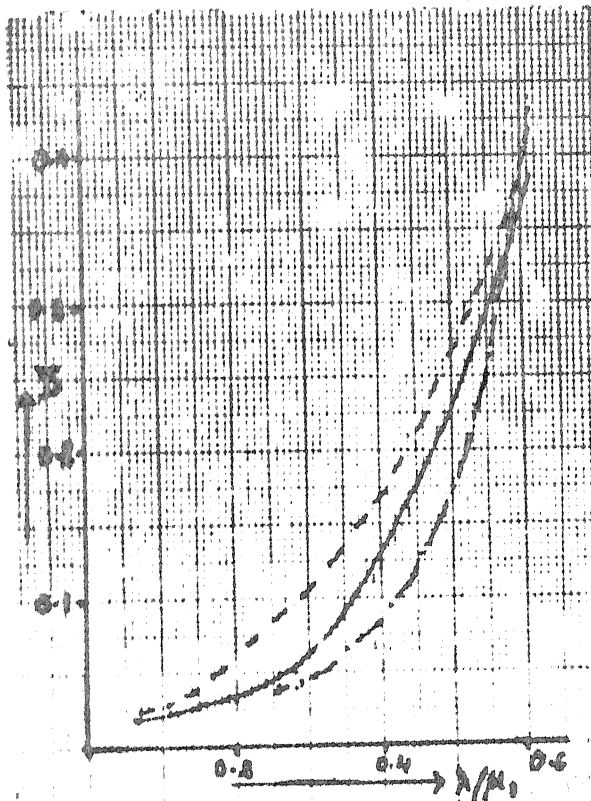
(a)



(b)

$$\mu_1 = 90.0, \mu_2 = 90.0$$

$$p_0 = 0.1, N = 25, W = 5, I = 5$$



(c)

Fig. 5.1 Variation of \bar{Z} , P , \bar{D} with λ

- (a) \bar{Z} vs λ/μ_1
 (b) P vs λ/μ_1
 (c) \bar{D} vs λ/μ_1

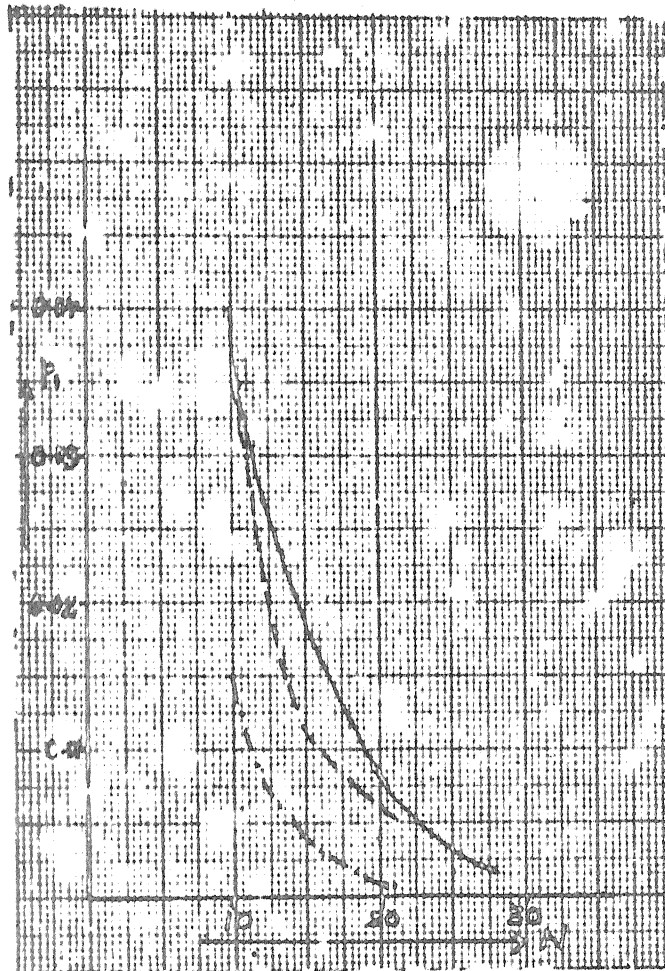
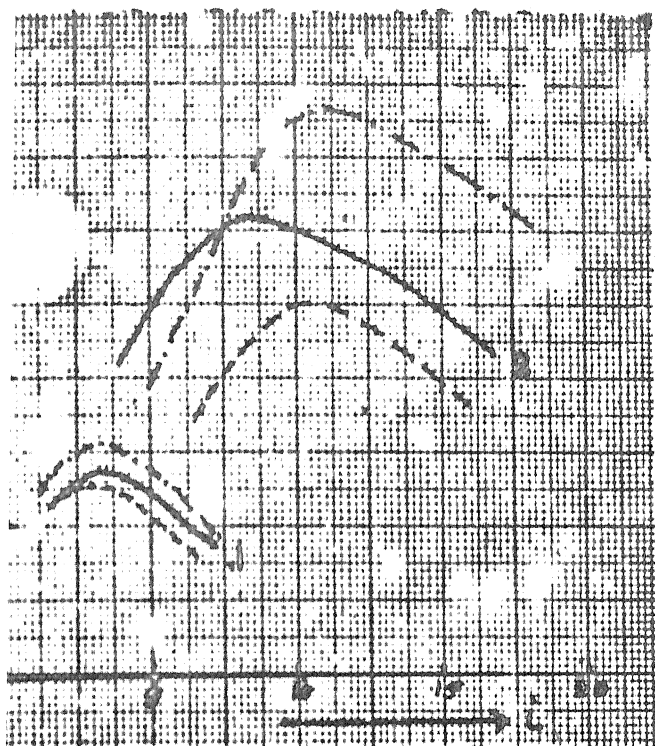


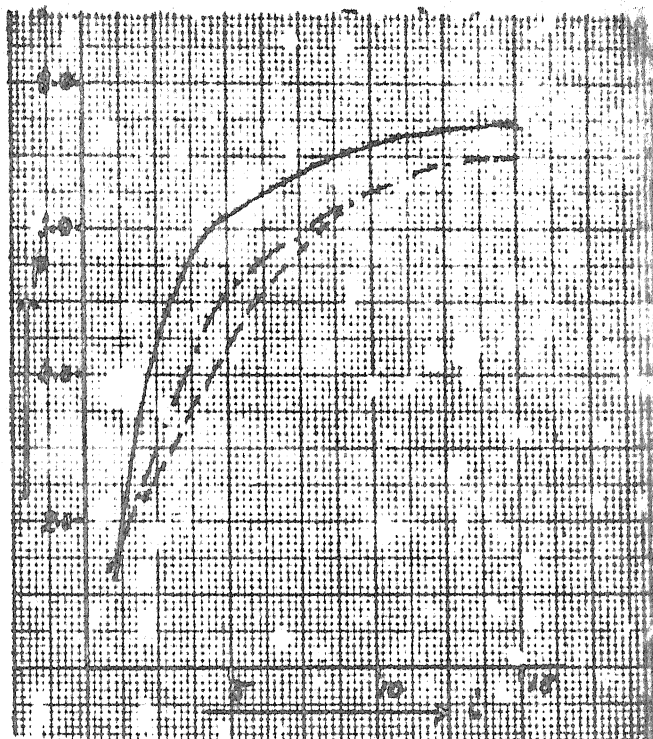
Fig. 3.2 Variation of buffer overflow probability vs N

$$\lambda = 40.0, \mu_1 = 95.0, \mu_2 = 100.0, p_0 = 0.1,$$

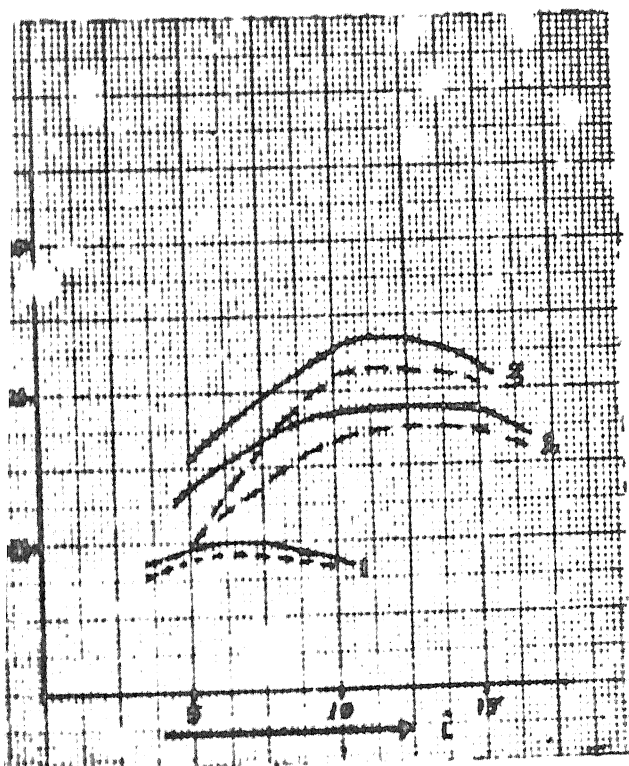
$$W = 5, l = 5$$



(a)



(b)



(c)

(a) 1. $\lambda = 30.0, \mu_1 = 50.0, \mu_2 = 50.0$
 $p_0 = 0.08, N = 15, W = 5$

(b) 2. $\lambda = 30.0, \mu_1 = 50.0, \mu_2 = 50.0$
 $p_0 = 0.01, N = 15, W = 5$

(b) $\lambda = 6.0, \mu_1 = 10.0, p_0 = 0$
 $N = 15, W = 5$

(c) $\lambda = 40.0, \mu_1 = 50.0, \mu_2 = 50.0$
 $N = 15, W = 5$

1. $p_0 = 0.05$

2. $p_0 = 0.01$

3. $p_0 = 0.005$

Fig. 3.4 Variation of power with 1

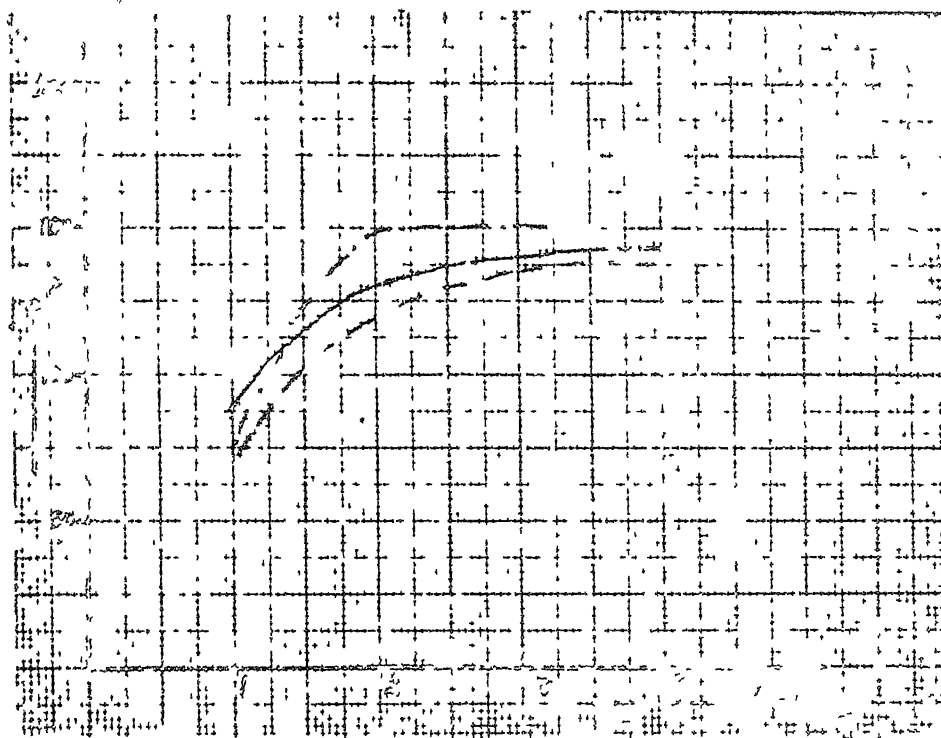


$$\lambda = 30.0, \mu_1 = 50.0, p_0 = 0.03, N = 15, W = 5, l = 5$$

Fig. 5.5 Variation of P with μ_2

REFERENCES

1. ACKR 80 Ackroyd M.H., 'Computing the Waiting Time Distribution for the G/G/1 Queue by Signal Processing Methods', IEEE Trans. Comm. COM-28, 1 (Jan 1980), 52-58.
2. BASK 75 Basket F., K.M. Chandy, R.R. Muntz and F.G. Palacios', 'Open, Closed and Mixed Network of Queues with Different Classes of Customers', JACM, vol. 22, 2 (April 1975), 248-260.
3. BAUM 79 Baum D., W. Lehmann-Bauerfield, R. Popescu-Zeletin and K. Ullmann, 'End-to-End Data Flow Analysis for Communication Networks', in Grange and Gien (eds), Proc. Int. Symp. on Flow Control in Computer Networks, Versailles, France, Feb. 1979, Amsterdam: North Holland, 321-404.
4. BENN 79 Bennett C.J., 'Supporting Transnet Bulk Data Transfer', in Grange and Gien (eds), Proc. Int. Symp. on Flow Control in Computer Networks, Versailles, France, Feb. 1979, Amsterdam: North Holland, 383-427.
5. BHAR 80 Bharath-Kumar K., 'Discrete-Time Queuing Systems and their Networks', IEEE Trans. Comm. COM-28, 2 (Feb. 1980), 260-263.
6. BHAR 81 Bharath-Kumar K. and J.M. Jaffe, 'A New Approach to Performance-Oriented Flow Control', IEEE Trans. Comm., COM-29, 4 (April 1980), 427-435.
7. BIAL 80 Bially T., B. Gold and S. Seneff, 'A Technique for Adaptive Voice Flow Control in Integrated Packet Networks', IEEE Trans. Comm., COM-28, 3 (March 1980), 325-333.
8. BULL 59 Bullington K. and J.M. Fraser, 'Engineering Aspects of TASI', Bell Syst. Tech. Journal, 38 (1959), 353-364.
9. BURK 56 Burke P.J., 'The Output of a Queuing System', Operations Research, 4 (1956), 699-704.



$$\lambda = 30.0, \mu_1 = 50.0, p_0 = 0.08, N = 15, w = 5, l = 5$$

Fig. 5.5 Variation of P with μ_2

REFERENCES

1. ACKR 80 Ackroyd M.H., 'Computing the Waiting Time Distribution for the G/G/1 Queue by Signal Processing Methods', IEEE Trans. Comm. COM-28, 1 (Jan 1980), 52-58.
2. BASK 75 Baske tt F., K.M. Chandy, R.R. Muntz and F.G. Palacios', Open, Closed and Mixed Network of Queues with Different Classes of Customers', JACM, vol. 22, 2 (April 1975), 248-260.
3. BAUM 79 Baum D., W. Lehmann-Bauerfield, R. Popesu-Zeletin and K. Ullmann, 'End-to-End Data Flow Analysis for Communication Networks', in Grange and Gien (eds), Proc. Int. Symp. on Flow Control in Computer Networks, Versailles, France, Feb. 1979, Amsterdam: North Holland, 321-404.
4. BENN 79 Bennett C.J., 'Supporting Transnet Bulk Data Transfer', in Grange and Gien (eds), Proc. Int. Symp. on Flow Control in Computer Networks, Versailles, France, Feb. 1979, Amsterdam: North Holland, 383-427.
5. BHAR 80 Bharath-Kumar K., 'Discrete-Time Queuing Systems and their Networks', IEEE Trans.Comm. COM-28, 2(Feb. 1980), 260-263.
6. BHAR 81 Bharath-Kumar K. and J.M. Jaffe, 'A New Approach to Performance-Oriented Flow Control', IEEE Trans. Comm., COM-29, 4 (April 1980), 427-435.
7. BIAL 80 Bially T., B. Gold and S. Seneff, 'A Technique for Adaptive Voice Flow Control in Integrated Packet Networks', IEEE Trans. Comm., COM-28, 3 (March 1980), 325-333.
8. BULL 59 Bullington K. and J.M. Fraser, 'Engineering Aspects of TASI', Bell Syst.Tech.Journal, 38 (1959), 353-364.
9. BURK 56 Burke P.J., 'The Output of a Queuing System', Operations Research, 4 (1956), 699-704.

10. BURK 72 Burke P.J., 'Output Processes and Tandem Queues', in Proc. Symp. on Computer-Communications Networks and Teletraffic, Polytechnic Institute of Brooklyn (1972), 419-428.
11. BURK 76 Burke P.J., 'Proof of a Conjecture on the Interarrival time Distribution in a M/M/1 Queue with Feedback', IEEE Trans. Comm., COM-24, 5 (May 1976), 575-576.
12. BUZE 73 Buzen J.P., 'Computational Algorithms for Closed Queuing Networks with Exponential Servers', CACM, 16, 9 (Sep. 1973), 527-531.
13. CHAN 75 Chandy K.M., U. Herzog and L.S. Woo, 'Parametric Analysis of Queuing Networks', IBM J. Res. and Dev., 19, 1 (Jan. 1975), 43-49.
14. COHE 80 Cohen D., 'Flow Control for Real Time Communications', ACM SIGCOMM, vol. 10, Nos. 1-2 (Jan/April 1980), 41-48.
15. COUR 80 Courtois P., 'The M/G/1 Finite Capacity Queue with Delays', IEEE Trans. Comm., COM-28, 2 (Feb. 1980), 165-174.
16. COX 55 Cox D.R., 'A use of Complex Probabilities in the Theory of Stochastic Processes', Proc. Cambridge Phil. Soc., 51, 2 (1955), 313-319.
17. DALE 68 Daley D.J., 'The Correlation Structure of the Output of Some Single Server-Queuing Systems', Ann. Math. Stat., 39, 3 (1968), 1007-1019.
18. DAVI 79 Davies D.W., D.C.A. Barber, W.L. Price and C.M. Solomonides, Computer Networks and Their Protocols, New York: John Wiley (1979).
19. DEO 79 Deo N., System Simulation with Digital Computer. New Delhi: Prentice-Hall of India (1979).
20. EDGE 79 Edge S.W., 'Comparison of the Hop-by-Hop and Endpoint Approaches to Network Interconnections', In Grange and Gien (eds.), Proc. Int. Symp. on Flow Control in Computer Networks, Versailles, France, Feb. 1979, Amsterdam: North-Holland (1979), 359-378.

21. FAYO 78 Fayolle G., 'An Analytical Evaluation of the Performance of the Send-and-Wait Protocol', IEEE Trans. Comm., COM-26, 3 (March 1978), 313-319.
22. FINC 58 Finch P.D., 'The Effect of the Size of the Waiting Room on a Simple Queue', J. Roy. Stat. Soc., Ser B., 20, 1 (1958), 182-186.
23. FINC 59 Finch P.D., 'The Output Process of the Queuing System M/G/1', J. Roy. Stat. Soc., Ser. B. 21, 2(1959), 375-380.
24. FISC 76 Fischer M.J. and T.C. Harris, 'A Model for Evaluating the Performance of an Integrated Circuit-and-Packet-Switched Multiplex Structure', IEEE Trans. Comm., COM-24, 2(Feb. 1976), 195-202.
25. FRED 79 Fredericks A.A. and G.A. Reisner, 'Approximations to Stochastic Service Systems with an Application to Retrial Model', Bell Syst. Tech. Journal, 58, 3 (March 1979), 557-576.
26. FRED 82 Fredericks A.A., 'A Class of Approximations for the Waiting Time Distribution in a GI/G/1 Queuing System', Bell Syst. Tech. Journal, 61, 3 (March 1982), 295-325.
27. GERH 82 Gerhauser H.L., 'Digital Speech Interpolation with Predicted Wordlength Assignment', IEEE Trans. Comm., COM-30, 4 (April 1982),
28. GIES 81 Giessler A., A. Jagemann, E. Maser and J.O. Hanle, 'Flow Control Based on Buffer Class', IEEE Trans. Comm., COM-29, 4 (April 1981), 436-443.
29. GOLD 77 Gold D., 'Digital Speech Networks', Proc. IEEE, 65, 12 (Dec. 1977), 1636-1658.
30. GORD 67a Gordon W.J. and G.F. Newell, 'Closed Queuing Systems with Exponential Servers', Operations Research, 15, 2 (1967), 254-265.
31. GORD 67b ----, 'Cyclic Queuing Systems with Restricted Length Queues', Operations Research, 15, 2 (1967) 266-277.

32. GROS 74 Gross D. and C.M. Harris, Fundamentals of Queuing Theory, New York: John Wiley(1974).
33. HARR 80 Harrington E.A., 'Voice/Data Integration Using Circuit Switched Networks', IEEE Trans. Comm., COM-28, 6 (June 1980), 781-793.
34. HILL 67 Hill F.S. and R.W. Boling, 'Finite Queues in Series with Exponential or Erlang Service Times - A Numerical Approach', Operations Research, 15, 2 (1967), 286-303.
35. HSU 76 Hsu J. and P.J. Burke, 'Behaviour of Tandem Buffers with Geometric Input and Markovian Output', IEEE Trans. Comm., COM-24, 3 (March 1976), 358-366.
36. IRLA 77 Irland M.I. and G. Pujolle, 'Models of Tandem Network of Queues Motivated by Packet Switching', RIA Laboria Research Report, No.256 (Oct. 1977).
37. JACK 57 Jackson J.R., 'Networks of Waiting Lines', Operations Research 5 (1957), 518-521.
38. JAFF 81 Jaffe J.M., 'Flow Control Power is Non-decentralizable', IEEE Trans. Comm., COM-29, 9 (Sep. 1981), 1301-1306.
39. JAYA 76 Jayant N.S., 'Average-and Median-based Smoothing Techniques for Digital Speech Quality in the Presence of Transmission Errors', IEEE Trans. Comm., COM-24, 9 (Sep. 1976), 1043-1045.
40. JEWE 67 Jewell, W.S., 'A Simple Proof of the Formula $L = W\lambda$ ', Operations Research, 15, 6 (1967), 1109-1116.
41. KAMO 80 Kamoun F. and L. Kleinrock, 'Analysis of Finite Storage in Computer Network Node Environment under General Traffic Conditions', IEEE Trans. Comm., COM-28, 7 (July 1980), 992-1003.
42. KAMO 81 Kamoun F., 'A Drop and Throttle Flow Control Policy for Computer Networks', IEEE Trans. Comm., COM-29, 4 (April 1981), 444-452.

43. KANA 78 Kanai L.N. and A.R.K. Sastry, 'Models for Channels with Memory and Their Applications to Error Control', Proc. IEEE, vol.66, 7 (July 1978), 724-743.
44. KAUF 81 Kaufman L., B. Gopinath and E.F. Wunderlich, 'Analysis of Packet Network Congestion Control Using Sparse Matrix Algorithm', IEEE Trans. Comm., COM-29, 4 (April 1981), 453-465.
45. KEND 51 Kendall D.G., 'Some Problems in the Theory of Queues', J.Roy.Stat.Soc., Ser B., 13, 2(1951), 151-173.
46. KEND 53 -----, 'Stochastic Processes Occurring in the Theory of Queues and their Analysis by the Method of Imbedded Markov Chain', Ann. of Math. Stat., 24 (1953), 338-354.
47. KERM 80 Kermani P. and L. Kleinrock, 'Dynamic Flow Control in Store-and-Forward Computer Networks', IEEE Trans. Comm., COM-28, 2 (Feb. 1980), 263-271.
48. KLEI 75 Kleinrock L., Queuing Systems Vol.I : Theory, New York: John Wiley (1975).
49. KLEI 76 -----, Queuing Systems Vol. II : Computer Applications, New York: John Wiley (1976).
50. KLEI 80a Kleinrock L. and P. Kermani, 'Static Flow Control in Store-and-Forward Computer Networks', IEEE Trans. Comm., COM-28, 2 (Feb. 1980), 271-278.
51. KLEI 80b Kleinrock L. and M. Gerla, 'Flow Control : A Comparative Survey', IEEE Trans. Comm., COM-28, 4 (April 1980), 553-574.
52. KING 62 Kingman J.F.C., 'On Queues in Heavy Traffic', J.Roy. Stat.Soc., Ser.B., 24, 2 (1962), 383-392.
53. KOBA 74a Kobayashi H., 'Application of the Diffusion Approximation to Queuing Networks I : Equilibrium Queue Distributions', JACM, vol. 21, 2 (April 1974), 316-328.

54. KOBA 74b -----, 'Application of the Diffusion Approximation to Queuing Networks II : Nonequilibrium Distributions and Applications to Computer Modelling', JACM, vol. 21, 3 (July, 1974), 459-469.
55. KOBA 75 Kobayashi H. and M. Reiser, 'On the Generalization of Job Routing Behaviour in a Queuing Network Model', IBM Research Report RC 5252, Yorktown Heights (1975).
56. KUEH 79 Kuehn P.J., 'Approximate Analysis of General Queuing Networks by Decomposition ', IEEE Trans. Comm., COM-27, 1(Jan.1979), 113-125.
57. KUEH 81 -----, 'Performance of ARQ-Protocols for HDX-transmission in Hierarchical Polling Systems', Performance Evaluation, vol.1, 1(1981), 20-30.
58. LAM 78 Lam S.S., 'A New Measure for Characterizing Data Traffic', IEEE Trans. Comm., COM-26, 1(Jan.1978), 137-140.
59. LAM 79 Lam S.S. and M. Reiser, 'Congestion Control of Store-and-Forward Networks by Input Buffer Limits - An Analysis', IEEE Trans. Comm., COM-27, 1(Jan. 1979), 127-133.
60. LIND 52 Lindley D.V., 'The Theory of Queues with a Single Server', Proc. Cambridge Phil.Soc., vol. 48 (1952), 277-289.
61. LITT 61 Little J.D.C., 'Proof of the Formula $L = W\lambda$ ', Operations Research, 9 (1961), 383.
62. MA 82 Ma J.S., 'On the Impact of HDLC Zero Insertion and Deletion on Link Utilization and Reliability', IEEE Trans. Comm., COM-30, 2 (Feb. 1982), 375-381.
63. MARS 68 Marshall K.T., 'Some Inequalities in Queuing', Operations Research, 16 '3(1968), 651-665.

64. MATS 81 Matsumoto J. and H. Mori, 'Flow Control in Packet-Switched Networks by Gradual Reduction of Virtual Calls', IEEE Trans. Comm., COM-29, 4(April 1981), 466-473.
65. MIRA 63 Mirasol N.M., 'The Output of a M/G/ ∞ Queue is Poisson', Operations Research, 11 (1963), 282-284.
66. PACK 78 Pack C.D., 'The Output of Multiserver Queuing Systems', Operations Research, 26,3 (1978), 492-509.
67. POUZ 81 Pouzin L., 'Methods, Tools and Observations on Flow Control in Packet-Switched Data Networks', IEEE Trans. Comm., COM-29, 4 (April 1981), 413-426.
68. REIS 79 Reiser M., 'A Queuing Network Analysis of Computer Communication Networks with Window Flow Control', IEEE Trans. Comm., COM-27, 8(Aug. 1979), 1199-1209.
69. REIS 80 Reiser M. and S.S. Lavenburg, 'Mean-value Analysis of Closed Multichain Queuing Networks', JACM, vol. 27, 2(April 1980), 313-322.
70. REIS 81a Reiser M., 'Mean-value Analysis and Convolution Method for Queue-Dependent Servers in Closed Queuing Networks', Performance Evaluation, vol. 1, 1(Jan. 1981), 7-18.
71. REIS 81b Reiser M., 'Admission Delays on Virtual Routes with Window Flow Control', in G. Pujolle (ed) Performance of Data Communication Systems and their Applications, Amsterdam: North-Holland (1981), 67-77.
72. ROSS 82 Ross M.J. and O.A. Mowaffi, 'Performance Analysis of Hybrid Switching Concepts for Integrated Voice/Data Communications', IEEE Trans. Comm., COM-30, 5(May 1982), 1073-1083.
73. SCHW 77 Schwartz M., Computer-Communication Network Design and Analysis, Englewood Cliffs, NJ: Prentice Hall (1977).

74. SCHW 80 Schwartz M. and T.E. Stearn, 'Routing Techniques Used in Computer Communication Networks', IEEE Trans. Comm., COM-28, 4(April 1980), 539-552.
75. SCHW 82 Schwartz M, 'Performance Analysis of the SNA Virtual Route Pacing Control', IEEE Trans. Comm., COM-30, 1(Jan. 1982), 172-184.
76. SCHWE 76 Schweitzer P. and S.S. Lam, 'Buffer Overflow in a Store-and-Forward Network Node', IBM J. Res. and Dev., vol. 20, 6 (Nov.1976), 542-550.
77. SPRO 78 Sproull R.F. and D. Cohen, 'High-level Protocols', Proc. IEEE, vol. 66, 11(Nov.1978), 1371-1385, 1371.
78. TAHA 71 Taha H.A., Operations Research : An Introduction, New York: MacMillan (1971).
79. WEIN 78 Weinstein C., 'Fractional Speech Loss and Talker Activity Model for TASI and for Packet-Switched Speech', IEEE Trans.Comm. COM-26, 8 (Aug. 1978), 1253-1257.
80. WEIN 80 Weinstein C., M.C. Malpass and M.J. Fisher, 'Traffic Performance of an Integrated Circuit-and-Packet Switched Multiplex structure', IEEE Trans. Comm., COM-28, 6(June 1980), 873-878.
81. WHIT 68 Whittle P., 'Equilibrium Distributions for an Open Migration Process', J.Appl. Probability, 5(1968), 567-571.
82. WHIT 75 White J.A., J.W. Schmidt and G.K.Bennett, Analysis of Queuing Systems, New York: Academic Press, 1975.
83. WONG 82 Wong J.W., J.P. Sauve and J.A. Field, 'A study of Fairness in Packet-Switching Networks', IEEE Trans. Comm., COM-30, 2 (Feb. 1982).

APPENDIX A

DETERMINATION OF PARAMETERS FOR AN H_2
DISTRIBUTION GIVEN MEAN AND COEFFICIENT
OF VARIATION

Let the mean be, $\bar{x} = 1/\mu$

and coeff. of variation = C_b .

The H_2 system, in Fig. A.1, is represented by two exponential processes in parallel, with α , $(1-\alpha)$ the probability of service at the respective server.

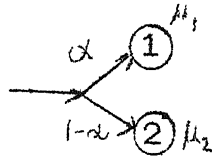


Fig. A.1

Then [WHIT 75, pp 153],

$$\mu_1 = 2\alpha\mu, \quad \mu_2 = 2(1-\alpha)\mu \quad (\text{A.1})$$

Now, the expectation,

$$\begin{aligned} E_t &= \alpha \cdot \frac{1}{\mu_1} + (1-\alpha) \frac{1}{\mu_2} \\ &= \frac{1}{\mu} \end{aligned} \quad (\text{A.2})$$

The 2nd moment is given by [KLEI 75],

$$\begin{aligned}\overline{t^2} &= \frac{d}{ds}^2 \left\{ \frac{\alpha \mu_1}{(s+\mu_1)} + \frac{(1-\alpha)\mu_2}{(s+\mu_2)} \right\}_{s=0} \\ &= \frac{2\alpha}{\mu_1^2} + \frac{2(1-\alpha)}{\mu_2^2}\end{aligned}\quad (A.3)$$

The variance,

$$\begin{aligned}V_t &= \frac{2\alpha}{\mu_1^2} + \frac{2(1-\alpha)}{\mu_2^2} - E_t^2 \\ &= \frac{2\alpha}{4\alpha^2\mu^2} + \frac{2(1-\alpha)}{4(1-\alpha)^2\mu^2} - \frac{1}{\mu^2} \\ &= \frac{1}{\mu^2} \left[\frac{(1-\alpha)}{2\alpha(1-\alpha)} + \frac{\alpha}{2\alpha(1-\alpha)} - 1 \right] \\ &= \frac{1}{\mu^2} \left[\frac{1}{2\alpha(1-\alpha)} - 1 \right] \\ \text{or, } V_t &= E_t^2 \left[\frac{1-2\alpha(1-\alpha)}{2\alpha(1-\alpha)} \right] \\ \text{or, } C_b^2 &= \frac{V_t}{E_t^2} = \frac{1-2\alpha(1-\alpha)}{2\alpha(1-\alpha)}\end{aligned}\quad (A.4)$$

To solve α in terms of C_b we have

$$2\alpha(1-\alpha) C_b^2 + 2\alpha(1-\alpha) - 1 = 0$$

$$\text{or, } 2\alpha^2 [1 + C_b^2] - 2\alpha [1 + C_b^2] + 1 = 0$$

$$\text{or, } \alpha^2 - \alpha + \frac{1}{2[1 + C_b^2]} = 0$$

This is solved to give,

$$\begin{aligned} \alpha &= \frac{1}{2} \left[1 \pm \sqrt{1 - \frac{4}{2(1 + C_b^2)}} \right] \\ &= \frac{1}{2} \left[1 \pm \sqrt{\frac{C_b^2 - 1}{C_b^2 + 1}} \right] \end{aligned}$$

Choosing the -ve sign to keep $\alpha < 0.5$ [WHIT 75]

we have,

$$\alpha = \frac{1}{2} \left[1 - \sqrt{\frac{C_b^2 - 1}{C_b^2 + 1}} \right] \quad (\text{A.5})$$

Queuing system analysis requires only the mean and variance. Thus any system with coeff. of variation, $C_b > 1$ can be given an H_2 representation by eqn. A.5. If approximation to n moments were necessary, an equivalent H_n system would have been required.

APPENDIX B

TRAFFIC INTENSITY FOR AN $H_2/M/1/\infty$ SYSTEM

Let the arrival process, assumed H_2 , have the transform,

$$A^*(s) = \frac{\alpha \mu_1}{(s + \mu_1)} + \frac{(1-\alpha) \mu_2}{(s + \mu_2)} \quad (B.1)$$

where $\mu_1 = 2\alpha \lambda = 2\alpha k\mu$

$$\mu_2 = 2(1-\alpha) \lambda = 2(1-\alpha)k\mu \quad (B.2)$$

(Refer to Appendix A).

where $1/\lambda$ is the mean interarrival time, the service time follows exponential distribution with mean $1/\mu$ and $k = \lambda/\mu$. k corresponds to the traffic intensity of $M/M/1$ system if arrival was poisson.

$$A^*(s) = \frac{s [\alpha \mu_1 + (1-\alpha) \mu_2] + \mu_1 \mu_2}{(s + \mu_1)(s + \mu_2)} \quad (B.3)$$

Now for a $GI/M/1$ system the traffic intensity, σ is such that (eq. 2.3.2.4)

$$\sigma = A^*(\mu - \mu\sigma)$$

This gives the state probabilities,

$$p_k = \sigma^k (1 - \sigma^k) \quad , \quad k = 0, 1, 2, \dots$$

$$\sigma = \frac{\mu (1-\sigma) [2\alpha^2 k\mu + 2 (1-\alpha)^2 k\mu] + 4 (1-\alpha)\alpha k^2 \mu^2}{[\mu (1-\sigma) + 2\alpha k\mu] [\mu(1-\sigma) + 2 (1-\alpha) k\mu]}$$

Cancelling μ^2 and letting $r = (1-\sigma)$,

$$\begin{aligned} r &= \frac{2 rk [\alpha^2 + (1-\alpha)^2] + 4 k^2 \alpha (1-\alpha)}{[r + 2\alpha k] [r + 2 (1-\alpha) k]} \\ &= \frac{2 rk [\alpha^2 + (1-\alpha)^2] + 4 k^2 \alpha (1-\alpha)}{r^2 + 2 kr + 4\alpha (1-\alpha) k^2} \end{aligned}$$

Rearranging and cancelling terms,

$$r^2 - r^3 - 2kr^2 - 4 r\alpha (1-\alpha)k^2 + 4 rk\alpha (1-\alpha) = 0$$

Cancelling r throughout and rearranging,

$$r^2 - [1 - 2k] r - 4\alpha (1-\alpha) k (1-k) = 0$$

$$r = \frac{[1-2k] \pm \sqrt{(1-2k)^2 + [16 k (1-k) \alpha (1-\alpha)]}}{2}$$

reverting back to σ ,

$$\sigma = \frac{1}{2} + k \mp \frac{1}{2} \sqrt{(1-2k)^2 + 16 k (1-k)\alpha (1-\alpha)}$$

We choose the - ve sign so that $\sigma < 1$

$$\text{or, } \sigma = k + \left[\frac{1}{2} - \frac{1}{2} \sqrt{(1-2k)^2 + 16 k (1-k) \alpha (1-\alpha)} \right] \quad (\text{B.4})$$

$$= k + H_2 c$$

where $H_2 c$ = correction term due to H_2 input, $k = \lambda/\mu$,
the traffic intensity for M/M/1 queue.

Thus given the coeff. of variation of a process, C_{ba} , we calculate [Appendix A]

$$\alpha = \frac{1}{2} - \frac{1}{2} \sqrt{\frac{C_{ba}^2 - 1}{C_{ba}^2 + 1}}$$

$$\text{and then } H_2^c = \frac{1}{2} [1 - \sqrt{(1 - 2K)^2 + 16k(1 - k)\alpha(1 - \alpha)}]$$

$$\text{when } \alpha = 0.5 \quad (B.5)$$

$$\begin{aligned} & \sqrt{(1 - 2k)^2 + 16k(1 - k)\alpha(1 - \alpha)} \\ & \equiv \sqrt{(1 - 2k)^2 + 4k(1 - k)} \\ & = \sqrt{1 - 4k + 4k^2 + 4k - 4k^2} = 1 \end{aligned}$$

or, $H_2^c = \frac{1}{2} - \frac{1}{2} = 0$, (as expected - this is the M/M/1 case).

when $\alpha = 0$

$$\begin{aligned} H_2^c &= \frac{1}{2} [1 - \sqrt{(1 - 2k)^2}] = \frac{1}{2} [1 - (1 - 2k)] \\ &= k \end{aligned}$$

Thus the upper bound for traffic intensity for a $H_2/M/1/\infty$ queue is twice that for M/M/1/ ∞ queue with same parameters.

Also,

$$H_2^c = \frac{1}{2} [1 - \sqrt{(1 - 2k)^2 + 16\alpha(1 - \alpha)k(1 - k)}]$$

for a constant α , let $\alpha(1 - \alpha) = b$

$$\begin{aligned} H_2^c &= \frac{1}{2} [1 - \sqrt{1 - 4k(1 - 4b) + 4k^2(1 - 4b)}] \\ &= \frac{1}{2} [1 - \sqrt{1 - 4k(1 - k)(1 - 4b)}] \quad (B.6) \end{aligned}$$

This is maximum at $k = 0.5$.

Thus the effect of this factor is more pronounced at moderate traffic.

MEAN AND VARIANCE OF EFFECTIVE SERVICE TIME

C.1 EQUIVALENT REPRESENTATION

In this section we derive the expression for the mean and variances of the effective service time at stage 1 of the model in chapter 3.

Referring to Fig.3.1, each retransmission means an exponential stage with parameter μ_1 in series with a timeout stage which is Erlang=1 (i exponential serves with parameter μ_1) in series. Effectively the retransmission stage is therefore Erlang - ($i + 1$). Letting $j = i + 1$, and using notations of chapter 3 (ref. sec 3.5) the overall system is shown in Fig c.1,

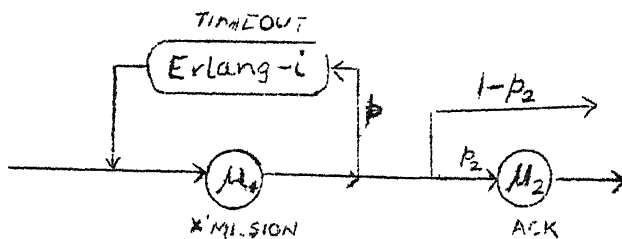


Fig. c.1

p is the probability of retransmission.

To allow calculations of the mean and variances of retransmissions only, overall service with no retransmissions (Sec. 3.4) we modify the representation to:

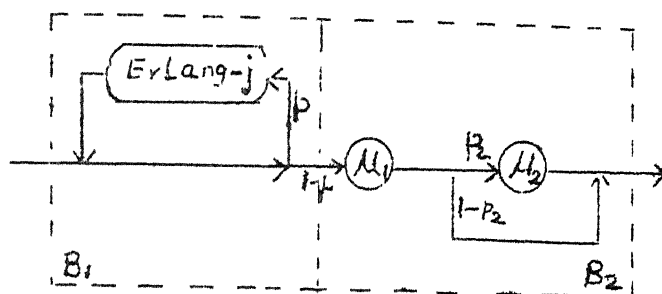


Fig. C.2

This does not change the moments of the overall service time pdf. We shall use the fact that the L-T. of pdf of servers in series is the product of the L-T.'s of individual servers. This leads to the variance of two servers in series is the sum of the individual variances. We shall calculate the overall variance by calculating variances of the individual stages B_1 and B_2 .

The L.T. of the retransmission stage,

$$B_1(s) = \frac{(1-p)}{1-p \left(\frac{\mu_1}{s + \mu_1} \right)^j} \quad (C.1)$$

Taking 1st and 2nd derivatives,

$$B_1'(s) = \frac{-(1-p) j p \mu_1^j \left(\frac{1}{s + \mu_1} \right)^{j+1}}{\left[1 - p \left(\frac{\mu_1}{s + \mu_1} \right)^j \right]^2} \quad (C.2)$$

$$\begin{aligned}
B_1'''(s) = - \left\{ - [1 - p \left(\frac{\mu_1}{s + \mu_1} \right)^J]^2 [(1-p)pj(j+1) \times \right. \\
\left. \mu_1^j \left(\frac{1}{s + \mu_1} \right)^{j+2}] - \right. \\
(1-p)jp\mu_1^j \left(\frac{1}{s + \mu_1} \right)^{j+1} 2[1-p \left(\frac{\mu_1}{s + \mu_1} \right)^J] \times \\
\left. [-jp\mu_1^j \left(\frac{1}{s + \mu_1} \right)^{j+1}] \right\} / \\
[1-p \left(\frac{\mu_1}{s + \mu_1} \right)^J]^4 \quad (C.3)
\end{aligned}$$

The mean,

$$\bar{s}_1 = -B_1'(0) = \frac{(1-p)jp}{\mu_1(1-p)^2} = \frac{jp}{\mu_1(1-p)} \quad (C.4)$$

The 2nd moment, given by $B''(0)$,

$$\overline{s_1^2} = \frac{jp(1+j+jp-p)}{\mu_1^2(1-p)^2} \quad (C.5)$$

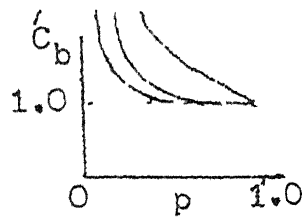
The variance,

$$\begin{aligned}
V(s_1) &= \overline{s_1^2} - \bar{s}_1^2 \\
&= \frac{jp(1+j-p)}{\mu_1^2(1-p)^2} \quad (C.6)
\end{aligned}$$

The coefficient of variation,

$$C_b(s_1) = \frac{\sqrt{V(s_1)}}{\bar{s}_1} = \sqrt{\frac{j+(1-p)}{jp}} \quad (C.77)$$

The general nature of C_b for the retransmission stage is shown below:



The second stage gives rise to,

$$B_2(s) = \frac{\mu_1(1-p_2)}{(s+\mu_1)^2} + p_2 \left(\frac{\mu_1}{s+\mu_1} \right) \left(\frac{\mu_2}{s+\mu_2} \right) \quad (C.8)$$

We calculate $B_2'(s)$ and $B_2''(s)$ by differentiation of

$$\text{eqn. C.7} \quad (C.9)$$

The mean,

$$\bar{s}_2 = -B_2'(0) = \frac{1}{\mu_1} + p_2 \frac{1}{\mu_2} \quad (C.10)$$

The 2nd moment,

$$\overline{s_2^2} = B_2''(0) = \frac{2}{\mu_1^2} + 2p_2 \frac{1}{\mu_1\mu_2} + 2p_2 \frac{1}{\mu_2^2} \quad (C.11)$$

from eqn C.10, we have ,

$$\overline{s_2^2} = \frac{1}{\mu_1^2} + 2p_2 \frac{1}{\mu_1\mu_2} + \frac{1}{\mu_2^2} \quad (\text{C.12})$$

the variance, from eqns. (C.11) and (C.12)

$$\begin{aligned} V(s_2) &= \overline{s_2^2} - \overline{s_2}^2 \\ &= \frac{1}{\mu_1^2} + 2p_2 \frac{1}{\mu_1\mu_2} - p_2^2 \frac{1}{\mu_2^2} \\ &= \frac{1}{\mu_1^2} + p_2(2-p_2) \frac{1}{\mu_2^2} \end{aligned} \quad (\text{C.13})$$

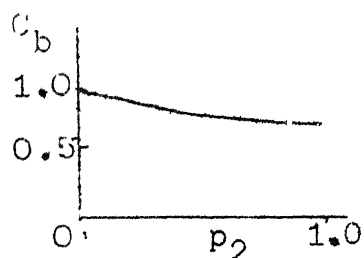
This gives,

$$C_b^2(s_2) = \frac{\mu_2^2 + p_2(2-p_2)\mu_1^2}{\mu_2^2 + 2p_2\mu_1\mu_2 + p_2^2\mu_1^2} \quad (\text{C.14})$$

When $\mu_1 = \mu_2 = \mu$,

$$C_b(s_2) = \frac{\sqrt{1+p_2(2-p_2)}}{(1+p_2)} \quad (\text{C.15})$$

The coeff. of variation decreases from 1.0 to $1/\sqrt{2}$, as shown,



The parameters of the overall service time at stage 1 of model are given by considering those of B_1 and B_2 in series.

The mean

$$\bar{S} = \bar{S}_1 + \bar{S}_2$$

or

$$\bar{S} = \frac{jp}{\mu_1(1-p)} + \frac{1}{\mu_1} + p_2 \frac{1}{\mu_2} \quad (C.16)$$

The variance

$$\begin{aligned} V(s) &= V(s_1) + V(s_2) \\ &= \frac{jp(1+j-p)}{\mu_1^2(1-p)^2} + \frac{1}{\mu_1^2} + p_2(2-p_2) \frac{1}{\mu_2^2} \end{aligned} \quad (C.17)$$

Therefore, the coefficient of variation, $C_b(s) = \frac{\sqrt{V(s)}}{\bar{S}}$.

The case $p = 0$ corresponds to no retransmission.

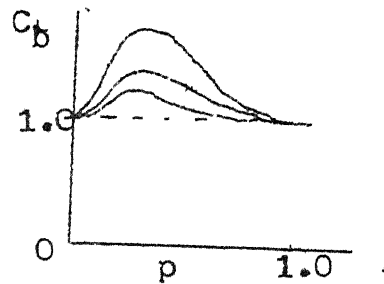
This is same as eqn. (C.14).

To find the coefficient of variation of server with retransmission (stage 1 of model in Chapter 3), we set $p_2 = 0$ in eqns. (C.16), and (C.17).

Therefore,

$$C_b = \frac{\sqrt{jp(j+1-p)+(1-p)^2}}{jp + (1-p)} \quad (C.18)$$

The general nature of the curves is plotted below.



This shows that the server's coefficient of variation deviates from 1.0 at a certain range of p values. Thus physical network nodes with low retransmission probabilities may be modelled as exponential.

APPENDIX D

THE PROBABILITY OF REDUNDANT TRANSMISSIONS

We base this calculation on the model and the notations of Chapter 3.

The probability of redundant transmission,

$$p_r = \sum_{k=1}^W \text{prob} (<K \text{ acknowledgements in Timeout}) \times \text{prob}(K \text{ packets waiting for ack.}) \quad (\text{D.1})$$

or,

$$p_r = \sum_{k=0}^{W-1} \text{prob}(K \text{ ack's in Time-out}) \times \text{prob} (>K \text{ waiting for ack.}) \quad (\text{D.2})$$

The last form is preferred because of ease of calculation.

Probability of K ack's in time-out,

$$p_1 = \frac{e^{-\mu_2 \text{ time-out}}}{K!} (\mu_2 \text{ time-out})^K \quad (\text{D.3})$$

because the acknowledgement process is poisson with parameter μ_2 .

$$\text{Time-out} = \frac{1}{\mu_1}$$

Hence,

$$p_1 = \frac{\left\{ e^{-\mu_2} \cdot \frac{1}{\mu_1} \right\} (\mu_2 \cdot \frac{1}{\mu_1})^K}{K!} \quad (D.4)$$

We now calculate, p_2 , the probability of greater than K packets waiting for Ack. If ρ_2 is the traffic intensity at stage 2, by analysis of a M/M/1/W system we have,

$$\begin{aligned} p_2 &= \sum_{j=k+1}^W \frac{(1-\rho_2) \rho_2^j}{(1-\rho_2^{W+1})} \\ &= \frac{(1-\rho_2)}{(1-\rho_2^{W+1})} \frac{\rho_2^{K+1} (1-\rho_2^{W-K})}{(1-\rho_2)} \end{aligned} \quad (D.5)$$

from (D.2), (D.4), (D.5) we have

$$p_r = \frac{e^{-\left(\frac{\mu_2}{\mu_1}\right)} \left(\frac{\mu_2}{\mu_1}\right)^{j+1}}{(1-\rho_2^{j+1})} \sum_{k=0}^{W-1} \left(\frac{\mu_2}{\mu_1}\right)^K \rho_2^{K+1} (1-\rho_2^{W-K}) \quad (D.6)$$

For $W = \infty$, a simple closed form solution is possible.

The probability of greater than K packets waiting,

$$\begin{aligned}
 p_2 &= \sum_{j=k+1}^{\infty} (1 - \rho_2) \rho_2^j \\
 &= \frac{(1 - \rho_2) \rho_2^{K+1}}{(1 - \rho_2)} = \rho_2^{K+1}
 \end{aligned} \tag{D.7}$$

or,

$$\begin{aligned}
 p_r &= \sum_{k=0}^{\infty} \frac{\left\{ e^{-i \frac{\mu_2}{\mu_1}} \right\} \left(i \frac{\mu_2}{\mu_1} \right)^K \rho_2^{K+1}}{K!} \\
 &= \left\{ e^{-i \frac{\mu_2}{\mu_1}} \right\} \rho_2 \sum_{k=0}^{\infty} \frac{\left(\rho_2 i \frac{\mu_2}{\mu_1} \right)^K}{K!} \\
 &= \rho_2 e^{-i \frac{\mu_2}{\mu_1}} e^{\rho_2 i \frac{\mu_2}{\mu_1}} \\
 p_r &= \rho_2 e^{-i \frac{\mu_2}{\mu_1}} (1 - \rho_2)
 \end{aligned} \tag{D.8}$$

NTRAL LIBRARY
no 11
Vo. 4 82773

SP-1982-M-GHO-SUC